

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ

**«БРАТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

УТВЕРЖДАЮ:

Проректор по учебной работе

\_\_\_\_\_ Е.И. Луковникова

« \_\_\_\_\_ » \_\_\_\_\_ 201\_\_ г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ**

**Б1.Б.12**

**НАПРАВЛЕНИЕ ПОДГОТОВКИ**

**09.03.02 Информационные системы и технологии**

**ПРОФИЛЬ ПОДГОТОВКИ**

**Информационные системы и технологии**

Программа академического бакалавриата

Квалификация (степень) выпускника: бакалавр

|  |           |
|--|-----------|
| <b>1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ .....</b> | <b>3</b>  |
| <b>2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ .....</b>   | <b>4</b>  |
| <b>3. РАСПРЕДЕЛЕНИЕ ОБЪЕМА ДИСЦИПЛИНЫ</b>  | <b>5</b>  |
| 3.1 Распределение объёма дисциплины по формам обучения.....  | 5         |
| 3.2 Распределение объёма дисциплины по видам учебных занятий и трудоемкости .....  | 5         |
| <b>4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ .....</b>  | <b>6</b>  |
| 4.1 Распределение разделов дисциплины по видам учебных занятий .....   | 6         |
| 4.2 Содержание дисциплины, структурированное по разделам и темам .....   | 8         |
| 4.3 Лабораторные работы.....   | 10        |
| 4.4 Семинары / практические занятия.....   | 11        |
| 4.5 Контрольные мероприятия: курсовой проект (курсовая работа), контрольная работа, РГР, реферат.....  | 11        |
| <b>5. МАТРИЦА СООТНЕСЕНИЯ РАЗДЕЛОВ УЧЕБНОЙ ДИСЦИПЛИНЫ К ФОРМИРУЕМЫМ В НИХ КОМПЕТЕНЦИЯМ И ОЦЕНКЕ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ .....</b>            | <b>12</b> |
| <b>6. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ</b>   | <b>13</b> |
| <b>7. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....</b>  | <b>13</b> |
| <b>8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО – ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ» НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ .....</b>                           | <b>14</b> |
| <b>9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ.....</b>  | <b>14</b> |
| 9.1. Методические указания для обучающихся по выполнению лабораторных работ/ семинаров / практических работ .....                                    | 15        |
| 9.2. Методические указания по выполнению курсового проекта (курсовой работы), контрольной работы, РГР, реферата .....                                | 47        |
| <b>10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ .....</b>                          | <b>49</b> |
| <b>11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ .....</b>                        | <b>49</b> |
| <b>Приложение 1. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине.....</b>                                   | <b>50</b> |
| <b>Приложение 2. Аннотация рабочей программы дисциплины .....</b>  | <b>55</b> |
| <b>Приложение 3. Протокол о дополнениях и изменениях в рабочей программе .....</b>   | <b>56</b> |

# 1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

## Вид деятельности выпускника

Дисциплина охватывает круг вопросов, относящихся к производственно-технологическому и сервисно-эксплуатационному видам профессиональной деятельности выпускника в соответствии с компетенциями и видами деятельности, указанными в учебном плане.

## Цель дисциплины

Изучение основ современных технологий программирования и получение практических навыков их реализации.

Задачами изучения дисциплины является изучение структуры и общих свойств информации и информационных процессов, общих принципов построения вычислительных устройств, а также систем обработки, хранения и передачи информации.

| Код компетенции | Содержание компетенций  | Перечень планируемых результатов обучения по дисциплине  |
|-----------------|---|--|
| 1               | 2   | 3  |
| ПК-15           | способность участвовать в работах по доводке и освоению информационных технологий в ходе внедрения и эксплуатации информационных систем   | <p><b>знать:</b></p> <ul style="list-style-type: none"> <li>– принципы обеспечения на стадии проектирования критериев качества программного средства облегчающих его доводку и освоение в ходе внедрения и эксплуатации;</li> </ul> <p><b>уметь:</b></p> <ul style="list-style-type: none"> <li>– организовать разработку программного средства в соответствии с приоритетами обеспечения качества;</li> </ul> <p><b>владеть:</b></p> <ul style="list-style-type: none"> <li>– навыками тестирования и отладки программного средства.</li> </ul> |
| ПК-17           | способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент, банковские системы, безопасность информационных систем, управление технологическими процессами, механика, техническая физика, энергетика, ядерная энергетика, силовая электроника, металлургия, строительство, транспорт, железнодорожный транспорт, связь, телекоммуникации, управление инфокоммуникациями, почтовая связь, химическая промышленность, сельское | <p><b>знать:</b></p> <ul style="list-style-type: none"> <li>– методы разработки программного средства с помощью компьютерных технологий;</li> </ul> <p><b>уметь:</b></p> <ul style="list-style-type: none"> <li>– создавать и модифицировать проект в изучаемом CASE-средстве с использованием имеющихся в нем элементов управления;</li> </ul> <p><b>владеть:</b></p> <ul style="list-style-type: none"> <li>– навыками объектно-ориентированного и визуального программирования.</li> </ul>  |

|       |   |  |
|-------|---|--|
|       | <p>хозяйство, текстильная и легкая промышленность, пищевая промышленность, медицинские и биотехнологии, горное дело, обеспечение безопасности подземных предприятий и производств, геология, нефтегазовая отрасль, геодезия и картография, геоинформационные системы, лесной комплекс, химико-лесной комплекс, экология, сфера сервиса, системы массовой информации, дизайн, медиаиндустрия, а также предприятия различного профиля и все виды деятельности в условиях экономики информационного общества</p> |  |
| ПК-32 | <p>способность адаптировать приложения к изменяющимся условиям функционирования</p>   | <p><b>знать:</b></p> <ul style="list-style-type: none"> <li>– методы объектного подхода к разработке программного средства;</li> </ul> <p><b>уметь:</b></p> <ul style="list-style-type: none"> <li>– применять изучаемые компьютерные технологии к адаптации программного средства к изменяющимся условиям функционирования;</li> </ul> <p><b>владеть:</b></p> <ul style="list-style-type: none"> <li>– навыками представления модели предметной области в терминах объектов.</li> </ul> |
| ОПК-1 | <p>владение широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий</p>   | <p><b>знать:</b></p> <ul style="list-style-type: none"> <li>– цели и этапы разработки программного средства;</li> </ul> <p><b>уметь:</b></p> <ul style="list-style-type: none"> <li>– создавать и модифицировать программный код в соответствии с правилами изучаемого языка программирования;</li> </ul> <p><b>владеть:</b></p> <ul style="list-style-type: none"> <li>– навыками структурного программирования.</li> </ul>   |

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина Б1.Б.12 «Технологии программирования» относится к базовой части.

Дисциплина «Технологии программирования» базируется на знаниях, полученных при изучении дисциплин «Информатика» и «Информационные технологии».

Основываясь на изучении перечисленных дисциплин, «Технологии программирования» представляет основу для изучения дисциплин: «Управление данными», «Программирование в информационных системах» и «Современное аппаратное обеспечение информационных систем».

Такое системное междисциплинарное изучение направлено на достижение требуемого ФГОС уровня подготовки по квалификации бакалавр.

### 3. РАСПРЕДЕЛЕНИЕ ОБЪЕМА ДИСЦИПЛИНЫ

#### 3.1. Распределение объема дисциплины по формам обучения

| Форма обучения                   | Курс | Семестр | Трудоемкость дисциплины в часах |                  |        |                     |                                  |                        | Контрольная работа, РГР | Вид промежуточной аттестации |
|----------------------------------|------|---------|---------------------------------|------------------|--------|---------------------|----------------------------------|------------------------|-------------------------|------------------------------|
|                                  |      |         | Всего часов (с экз.)            | Аудиторных часов | Лекции | Лабораторные работы | Семинары<br>Практические занятия | Самостоятельная работа |                         |                              |
| 1                                | 2    | 3       | 4                               | 5                | 6      | 7                   | 8                                | 9                      | 10                      | 11                           |
| Очная                            | 2,3  | 4,5     | 288                             | 123              | 53     | 70                  | -                                | 129                    | кр4,<br>РГР5            | зачёт,<br>экзамен            |
| Заочная                          | 2,3  | -       | 288                             | 38               | 12     | 22                  | 4                                | 237                    | кр2,<br>РГР3            | зачёт,<br>экзамен            |
| Заочная<br>(ускоренное обучение) | -    | -       | -                               | -                | -      | -                   | -                                | -                      | -                       | -                            |
| Очно-заочная                     | -    | -       | -                               | -                | -      | -                   | -                                | -                      | -                       | -                            |

#### 3.2. Распределение объема дисциплины по видам учебных занятий и трудоемкости

| Вид учебных занятий  | Трудоемкость (час.) | в т.ч. в интерактивной, активной, инновационной формах, (час.) | Распределение по семестрам, час |     |
|--|---------------------|--|---------------------------------|-----|
|  |                     |  | 4                               | 5   |
| 1  | 2                   | 3  | 4                               | 5   |
| <b>I. Контактная работа обучающихся с преподавателем (всего)</b> | 123                 | 12   | 72                              | 51  |
| Лекции   | 53                  | 8  | 36                              | 17  |
| Лабораторные работы  | 70                  | 4  | 36                              | 34  |
| Контрольная работа   | +                   | -  | +                               | -   |
| Расчетно-графическая работа                                      | +                   | -  | -                               | +   |
| Групповые консультации   | +                   | -  | +                               | +   |
| <b>II. Самостоятельная работа обучающихся</b>                    | 129                 | -  | 108                             | 21  |
| Подготовка к лабораторным работам                                | 60                  | -  | 52                              | 7   |
| Выполнение контрольной работы                                    | 30                  | -  | 30                              | -   |
| Выполнение расчетно-графической работы                           | 4                   | -  | -                               | 4   |
| Подготовка к зачету  | 25                  | -  | 26                              | -   |
| Подготовка к экзамену в течение семестра                         | 10                  | -  | -                               | 10  |
| <b>III. Промежуточная аттестация</b> зачет, экзамен              | 36                  | -  | -                               | 36  |
| Общая трудоемкость дисциплины .. час.                            | 288                 | -  | 180                             | 108 |
| зач. ед.   | 8                   | -  | 5                               | 3   |

## 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1. Распределение разделов дисциплины по видам учебных занятий

- для очной формы обучения:

| № раздела и темы | Наименование раздела и тема дисциплины   | Трудоемкость, (час.) | Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость; (час.) |                     |                                    |
|------------------|--|----------------------|---|---------------------|------------------------------------|
|                  |  |                      | учебные занятия   |                     | самостоятельная работа обучающихся |
|                  |  |                      | лекции  | лабораторные работы |                                    |
| 1                | 2  | 3                    | 4   | 5                   | 6                                  |
| <b>1.</b>        | <b>Программное средство (ПС)</b>   | <b>46</b>            | <b>6</b>  | <b>10</b>           | <b>30</b>                          |
| 1.1.             | Надежное ПС как продукт технологии программирования.                           | 18                   | 2   | 4                   | 12                                 |
| 1.2.             | Источники ошибок в ПС.   | 10                   | 2   | 2                   | 6                                  |
| 1.3.             | Общие принципы разработки ПС.  | 18                   | 2   | 4                   | 12                                 |
| <b>2.</b>        | <b>Этапы разработки ПС.</b>  | <b>134</b>           | <b>30</b>   | <b>26</b>           | <b>78</b>                          |
| 2.1.             | Внешнее описание ПС.   | 11                   | 3   | 2                   | 6                                  |
| 2.2.             | Функциональное описание ПС.  | 11                   | 3   | 2                   | 6                                  |
| 2.3.             | Архитектура ПС.  | 19                   | 3   | 4                   | 12                                 |
| 2.4.             | Структура программы и модульное программирование.                              | 11                   | 3   | 2                   | 6                                  |
| 2.5.             | Программный модуль.  | 19                   | 3   | 4                   | 12                                 |
| 2.6.             | Тестирование и отладка ПС.   | 11                   | 3   | 2                   | 6                                  |
| 2.7.             | Обеспечение функциональности и надежности ПС.                                  | 19                   | 3   | 4                   | 12                                 |
| 2.8.             | Обеспечение качества ПС.   | 11                   | 3   | 2                   | 6                                  |
| 2.9.             | Документирование ПС.   | 11                   | 3   | 2                   | 6                                  |
| 2.10             | Управление разработкой и аттестация ПС.  | 11                   | 3   | 2                   | 6                                  |
| <b>3.</b>        | <b>Компьютерная поддержка разработки ПС.</b>                                   | <b>31</b>            | <b>6</b>  | <b>16</b>           | <b>9</b>                           |
| 3.1.             | Объектный подход к разработке ПС.  | 9                    | 2   | 4                   | 3                                  |
| 3.2.             | Методы проектирования ПС   | 9                    | 2   | 4                   | 3                                  |
| 3.3.             | Case-средства  | 13                   | 2   | 8                   | 3                                  |
| <b>4.</b>        | <b>Разработка интерфейса ПС.</b>   | <b>41</b>            | <b>11</b>   | <b>18</b>           | <b>12</b>                          |
| 4.1.             | Проектирование человеко-машинного интерфейса.                                  | 6                    | 2   | 2                   | 2                                  |
| 4.2.             | Психология HCI.  | 4                    | 1   | 2                   | 1                                  |
| 4.3.             | Психология цветовосприятия   | 4                    | 1   | 2                   | 1                                  |
| 4.4.             | Визуальные атрибуты отображаемой информации. Композиция и организация экрана.  | 6                    | 2   | 2                   | 2                                  |
| 4.5.             | Разработка структуры диалога.  | 4                    | 1   | 2                   | 1                                  |
| 4.6.             | Окна как элементы графического интерфейса. Модели интерфейса.                  | 4                    | 1   | 2                   | 1                                  |
| 4.7.             | Объектно-ориентированный подход к проектированию интерфейса.                   | 4                    | 1   | 2                   | 1                                  |
| 4.8.             | Юзабилити-тестирование   | 5                    | 1   | 2                   | 2                                  |
| 4.9.             | Стандарты на интерфейсы. Средства проектирования пользовательских интерфейсов. | 4                    | 1   | 2                   | 1                                  |
|                  | <b>ИТОГО</b>   | <b>252</b>           | <b>53</b>   | <b>70</b>           | <b>129</b>                         |

- для заочной формы обучения:

| № раздела и темы | Наименование раздела и тема дисциплины   | Трудоемкость, (час.) | Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость; (час.) |                     |                      |                                    |
|------------------|--|----------------------|---|---------------------|----------------------|------------------------------------|
|                  |  |                      | учебные занятия   |                     |                      | самостоятельная работа обучающихся |
|                  |  |                      | лекции  | лабораторные работы | практические занятия |                                    |
| 1                | 2  | 3                    | 4   | 5                   | 6                    | 7                                  |
| <b>1.</b>        | <b>Программное средство (ПС)</b>   | <b>57</b>            | <b>2</b>  | <b>4</b>            |                      | <b>51</b>                          |
| 1.1.             | Надежное ПС как продукт технологии программирования.                           | 18,7                 | 0,7   | 1                   |                      | 17                                 |
| 1.2.             | Источники ошибок в ПС.   | 18,7                 | 0,7   | 1                   |                      | 17                                 |
| 1.3.             | Общие принципы разработки ПС.  | 19,6                 | 0,6   | 2                   |                      | 17                                 |
| <b>2.</b>        | <b>Этапы разработки ПС.</b>  | <b>191</b>           | <b>6</b>  | <b>10</b>           | <b>4</b>             | <b>171</b>                         |
| 2.1.             | Внешнее описание ПС.   | 19,6                 | 0,6   | 1                   |                      | 18                                 |
| 2.2.             | Функциональное описание ПС.  | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| 2.3.             | Архитектура ПС.  | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| 2.4.             | Структура программы и модульное программирование.                              | 22,6                 | 0,6   | 1                   | 4                    | 17                                 |
| 2.5.             | Программный модуль.  | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| 2.6.             | Тестирование и отладка ПС.   | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| 2.7.             | Обеспечение функциональности и надежности ПС.                                  | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| 2.8.             | Обеспечение качества ПС.   | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| 2.9.             | Документирование ПС.   | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| 2.10             | Управление разработкой и аттестация ПС.  | 18,6                 | 0,6   | 1                   |                      | 17                                 |
| <b>3.</b>        | <b>Компьютерная поддержка разработки ПС.</b>                                   | <b>13</b>            | <b>2</b>  | <b>5</b>            |                      | <b>6</b>                           |
| 3.1.             | Объектный подход к разработке ПС.  | 3,7                  | 0,7   | 1                   |                      | 2                                  |
| <b>1</b>         | <b>2</b>   | <b>3</b>             | <b>4</b>  | <b>5</b>            | <b>6</b>             |                                    |
| 3.2.             | Методы проектирования ПС   | 4,7                  | 0,7   | 2                   |                      | 2                                  |
| 3.3.             | Case-средства  | 4,6                  | 0,6   | 2                   |                      | 2                                  |
| <b>4.</b>        | <b>Разработка интерфейса ПС.</b>   | <b>14</b>            | <b>2</b>  | <b>3</b>            |                      | <b>9</b>                           |
| 4.1.             | Проектирование человеко-машинного интерфейса.                                  | 1,7                  | 0,3   | 0,4                 |                      | 1                                  |
| 4.2.             | Психология HCI.  | 1,6                  | 0,2   | 0,4                 |                      | 1                                  |
| 4.3.             | Психология цветовосприятия   | 1,6                  | 0,2   | 0,4                 |                      | 1                                  |
| 4.4.             | Визуальные атрибуты отображаемой информации. Композиция и организация экрана.  | 1,7                  | 0,3   | 0,4                 |                      | 1                                  |
| 4.5.             | Разработка структуры диалога.  | 1,6                  | 0,2   | 0,4                 |                      | 1                                  |
| 4.6.             | Окна как элементы графического интерфейса. Модели интерфейса.                  | 1,6                  | 0,2   | 0,4                 |                      | 1                                  |
| 4.7.             | Объектно-ориентированный подход к проектированию интерфейса.                   | 1,3                  | 0,2   | 0,1                 |                      | 1                                  |
| 4.8.             | Юзабилити-тестирование   | 1,6                  | 0,2   | 0,4                 |                      | 1                                  |
| 4.9.             | Стандарты на интерфейсы. Средства проектирования пользовательских интерфейсов. | 1,3                  | 0,2   | 0,1                 |                      | 1                                  |
|                  | <b>ИТОГО</b>   | <b>275</b>           | <b>12</b>   | <b>22</b>           | <b>4</b>             | <b>237</b>                         |

#### 4.2. Содержание дисциплины, структурированное по разделам и темам

| <i>№ раздела и темы</i> | <i>Наименование раздела и темы дисциплины</i>       | <i>Содержание лекционных занятий</i>   | <i>Вид занятия в интерактивной, активной, инновационной формах, (час.)</i> |
|-------------------------|---|--|--|
| <i>1</i>                | <i>2</i>  | <i>3</i>   | <i>4</i>   |
| <b>1.</b>               | <b>Программное средство (ПС)</b>                    |  |  |
| 1.1.                    | Надежное ПС как продукт технологии программирования | Программа как формализованное описание процесса обработки данных. Программное средство. Неконструктивность понятия правильной программы. Надежность программного средства. Технология программирования как технология разработки надежных программных средств. Технология программирования и информатизация общества.                              | Лекция-дискуссия, (0,8 час.)   |
| 1.2.                    | Источники ошибок в ПС.                              | Интеллектуальные возможности человека. Неправильный перевод как причина ошибок в программных средствах. Модель перевода. Основные пути борьбы с ошибками.  | Лекция-дискуссия, (0,8 час.)   |
| 1.3                     | Общие принципы разработки ПС.                       | Специфика разработки программных средств. Жизненный цикл программного средства. Понятие качества программного средства. Обеспечение надежности - основной мотив разработки программных средств. Методы борьбы со сложностью. Обеспечение точности перевода. Преодоление барьера между пользователем и разработчиком. Контроль принимаемых решений. | Лекция-дискуссия, (0,8 час.)   |
| <b>2.</b>               | <b>Этапы разработки ПС</b>                          |  |  |
| 2.1.                    | Внешнее описание ПС.                                | Назначение внешнего описания программного средства и его роль в обеспечении качества программного средства. Определение требований к программному средству. Спецификация качества программного средства. Функциональная спецификация программного средства. Методы контроля внешнего описания программного средства.                               | Лекция-дискуссия, (0,8 час.)   |
| 2.2.                    | Функциональное описание ПС.                         | Основные подходы к спецификации семантики функций. Семантика функции. Табличный подход (Метод таблиц решений). Алгебраический подход. Логический подход.   | Лекция-дискуссия, (0,8 час.)   |
| 2.3.                    | Архитектура ПС.                                     | Понятие архитектуры программного средства. Основные классы архитектур программных средств. Архитектурные функции. Контроль архитектуры программных средств.  | Лекция-дискуссия, (0,8 час.)   |
| 2.4.                    | Структура программы и модульное программирование.   | Цель модульного программирования. Основные характеристики программного модуля. Методы разработки структуры программы. Контроль структуры программы.  | Лекция-дискуссия, (0,8 час.)   |
| 2.5.                    | Программный модуль.                                 | Порядок разработки программного модуля. Структурное программирование. Пошаговая детализация и понятие о псевдокоде. Контроль программного модуля.  | Лекция-дискуссия, (0,8 час.)   |
| 2.6.                    | Тестирование и                                      | Основные понятия. Принципы и виды отладки про-   | Лекция-  |



|           |   |  |                              |
|-----------|---|--|------------------------------|
|           | отладка ПС.                                   | граммного средства. Общие рекомендации по организации отладки ПС. Автономная отладка программного средства. Комплексная отладка программного средства.   | дискуссия, (0,8 час.)        |
| 2.7.      | Обеспечение функциональности и надежности ПС. | Обеспечение завершенности программного средства. Обеспечение точности программного средства. Обеспечение автономности программного средства. Обеспечение устойчивости программного средства. Обеспечение защищенности программных средств.   | Лекция-дискуссия, (0,8 час.) |
| 2.8.      | Обеспечение качества ПС.                      | Общая характеристика процесса обеспечения качества ПС. Обеспечение легкости применения ПС. Обеспечение эффективности ПС. Обеспечение сопровождаемости ПС. Обеспечение мобильности ПС.  | -                            |
| 2.9.      | Документирование ПС.                          | Документация, создаваемая и используемая в процессе разработки программных средств. Пользовательская документация программных средств. Документация по сопровождению программных средств.  | -                            |
| 2.10      | Управление разработкой и аттестация ПС.       | Назначение и процессы управления разработкой программного средства. Структура управления разработкой программных средств. 13.3. Планирование и составление расписаний по разработке ПС. Аттестации программного средства.  | -                            |
| <b>3.</b> | <b>Компьютерная поддержка разработки ПС</b>   |  |                              |
| 3.1.      | Объектный подход к разработке ПС.             | Объекты и отношения в программировании. Сущность объектного подхода к разработке программных средств. Особенности объектного подхода к разработке внешнего описания программного средства. Особенности объектного подхода на этапе конструирования программного средства. Особенности объектного подхода на этапе кодирования программного средства.   | -                            |
| 3.3.      | Методы проектирования ПС                      | Процессы ЖЦ ПС. Процесс разработки. CASE-средства. CASE-технология. Методологии IDEF.  | -                            |
| 3.2.      | CASE-средства                                 | Инструменты разработки программных средств. Инструментальные среды разработки и сопровождения программных средств и принципы их классификации. Основные классы инструментальных сред разработки и сопровождения программных средств. Инструментальные среды программирования. Понятие компьютерной технологии разработки программных средств и ее рабочие места. Инструментальные системы технологии программирования. | -                            |
| <b>4.</b> | <b>Разработка интерфейса ПС</b>               |  |                              |
| 4.1.      | Проектирование человеко-машинного интерфейса. | Понятия и определения. Подходы к проектированию. Методологии разработки. Стандартизация. Рекомендации по проектированию пользовательского интерфейса. Этапы разработки пользовательского интерфейса. Определение требований к разработке. Методы анализа пользователей. Сценарии использования. Сортировка карточек.   | -                            |
| 4.2.      | Психология HCI.                               | Психология пользователя. Восприятие и внимание человека. Информационные процессы человека: память и познание. Мышление.  | -                            |
| 4.3.      | Психология цветовосприятия                    | Цвет как визуальный атрибут отображаемой информации. Характеристики цветов. Ощущение и восприятие цвета.   | -                            |

|      |  |   |   |
|------|--|---|---|
| 4.4. | Визуальные атрибуты отображаемой информации. Композиция и организация экрана.  | Визуальные атрибуты. Методы выделения информации. Композиция и организация. Шрифт как атрибут визуального отображения информации. «Многомерность» экрана. Пространственное размещение визуальных элементов. Визуальные объекты. | - |
| 4.5. | Разработка структуры диалога.  | Выбор структуры диалога. Управление данными при диалоге. Разработка сценария диалога. Описание структуры диалога с помощью сети переходов.  | - |
| 4.6. | Окна как элементы графического интерфейса. Модели интерфейса.                  | Модели построения интерфейса. Пиктограммы. Окна. Их виды и структура. Рабочая область – контейнер. Рабочая книга. Проект.   | - |
| 4.7. | Объектно-ориентированный подход к проектированию интерфейса.                   | Концепция интерфейса, управляемого данными. Перспективы эволюции интерфейсов – деятельностно-ориентированные интерфейсы.  | - |
| 4.8. | Юзабилити-тестирование   | Подбор пользователей. Методы оценки. Цели и задачи тестирования. Надежность и достоверность результатов тестирования.   | - |
| 4.9. | Стандарты на интерфейсы. Средства проектирования пользовательских интерфейсов. | Стандартизация пользовательского интерфейса. Эволюция пользовательского интерфейса. Инструментальные средства создания пользовательского интерфейса.  | - |

### 4.3. Лабораторные работы

| <i>№ п/п</i> | <i>Номер раздела дисциплины</i> | <i>Наименование тем лабораторных работ</i> | <i>Объем (час.)</i> | <i>Вид занятия в интерактивной, активной, инновационной формах, (час.)</i> |
|--------------|---------------------------------|--|---------------------|--|
| <i>1</i>     | <i>2</i>                        | <i>3</i>                                   | <i>4</i>            | <i>5</i>   |
| 1.           | 1.-2.                           | Целочисленные вычисления, CRT, GraphABC    | 4                   | Работа в малых группах (0,5 час.)  |
| 2.           | 1.-2.                           | Управляющие конструкции                    | 4                   | Работа в малых группах (0,5 час.)  |
| 3.           | 1.-2.                           | Подпрограммы и функции                     | 4                   | Работа в малых группах (0,5 час.)  |
| 4.           | 1.-2.                           | Массивы                                    | 6                   | Работа в малых группах (0,5 час.)  |
| 5.           | 1.-2.                           | Символы и строки. Файлы                    | 4                   | Работа в малых группах (0,5 час.)  |
| 6.           | 1.-2.                           | Пользовательские типы данных               | 4                   | Работа в малых группах (0,5 час.)  |
| 7.           | 1.-2.                           | Динамические структуры данных              | 6                   | Работа в малых группах (0,5 час.)  |
| 8.           | 1.-2.                           | Объекты                                    | 4                   | Работа в малых группах (0,5 час.)  |
| 9.           | 3.-4.                           | Первый проект                              | 2                   | -  |
| 10.          | 3.-4.                           | Внесение изменений и настройка проекта     | 2                   | -  |
| 11.          | 3.-4.                           | Form, Label, Button                        | 4                   | -  |
| 12.          | 3.-4.                           | Проверочная работа №1                      | 2                   | -  |

|              |       |   |           |          |
|--------------|-------|---|-----------|----------|
| 13.          | 3.-4. | CheckBox, RadioButton и ComboBox            | 2         | -        |
| 14.          | 3.-4. | ListBox, BitButton и Image                  | 2         | -        |
| 15.          | 3.-4. | Timer, UpDown и ProgressBar                 | 2         | -        |
| 16.          | 3.-4. | Panel, SpeedButton и StatusBar              | 2         | -        |
| 17.          | 3.-4. | Проверочная работа №2                       | 2         | -        |
| 18.          | 3.-4. | Memo, MainMenu, OpenFileDialog и SaveDialog | 2         | -        |
| 19.          | 3.-4. | TaskDialog, FileOpenDialog и FileSaveDialog | 2         | -        |
| 20.          | 3.-4. | Проверочная работа №3                       | 2         | -        |
| 21.          | 3.-4. | Графика. Часть 1                            | 2         | -        |
| 22.          | 3.-4. | Графика. Часть 2                            | 2         | -        |
| 23.          | 3.-4. | Графика. Часть 3                            | 2         | -        |
| 24.          | 3.-4. | Проверочная работа №4                       | 2         | -        |
| <b>ИТОГО</b> |       |   | <b>70</b> | <b>4</b> |

#### 4.4. Семинары / практические занятия

Учебным планом не предусмотрены.

#### 4.5. Контрольные мероприятия: контрольная работа, РГР

##### Контрольная работа

Цель: закрепление практических знаний о принципах структурного программирования.

Структура: титульный лист; содержание; введение; основные разделы работы; заключение; список использованных источников; приложение.

Основная тематика: контрольная работа выполняется по варианту, выдаваемому преподавателем.

Рекомендуемый объем: 10-15 страниц.

Выдача задания и прием контрольной работы проводится в соответствии с календарным учебным графиком.

| Оценка     | Критерии оценки контрольной работы  |
|------------|---|
| зачтено    | соответствие требованиям по структурному содержанию и объему работы; правильность выполнения задания; правильность решения практических заданий, самостоятельность выполнения; отсутствие стилистических ошибок; уверенное владение материалом при устной защите.                     |
| не зачтено | несоответствие требованиям по структурному содержанию и объему работы; неправильность выполнения задания; наличие ошибок в выполнении практических заданий; отсутствие самостоятельности выполнения; наличие стилистических ошибок; отсутствие владения материалом при устной защите. |

##### РГР

Цель: формирование практических знаний о базовых принципах разработки графического интерфейса пользователя.

Структура: титульный лист; содержание; введение; основные разделы работы; заключение; приложение.

Основная тематика: РГР выполняется по варианту, выдаваемому преподавателем.

Рекомендуемый объем: 10-15 страниц.

Выдача задания и прием РГР проводится в соответствии с календарным учебным графиком.

| Оценка     | Критерии оценки РГР  |
|------------|--|
| зачтено    | соответствие требованиям по структурному содержанию и объему работы; правильность выполнения задания, сопровождающегося рисунками, таблицами, диаграммами; правильность решения практических заданий, самостоятельность выполнения; отсутствие стилистических ошибок; уверенное владение материалом при устной защите.                 |
| не зачтено | несоответствие требованиям по структурному содержанию и объему работы; неправильность выполнения задания, небрежное выполнение рисунков, таблиц, диаграмм; наличие ошибок в выполнении практических заданий; отсутствие самостоятельности выполнения; наличие стилистических ошибок; отсутствие владения материалом при устной защите. |

**5. МАТРИЦА СООТНЕСЕНИЯ РАЗДЕЛОВ УЧЕБНОЙ ДИСЦИПЛИНЫ К ФОРМИРУЕМЫМ В НИХ КОМПЕТЕНЦИЯМ И ОЦЕНКЕ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

| <i>Компетенции</i><br><br><i>№, наименование<br/>разделов дисциплины</i> | <i>Кол-<br/>во<br/>часов</i> | <i>Компетенции</i> |           |           |            | $\Sigma$<br><i>комп.</i> | <i>t<sub>ср</sub>, час</i> | <i>Вид<br/>учебной<br/>работы</i> | <i>Оценка<br/>результатов</i> |
|--|------------------------------|--------------------|-----------|-----------|------------|--------------------------|----------------------------|-----------------------------------|-------------------------------|
|  |                              | <i>ПК</i>          |           |           | <i>ОПК</i> |                          |                            |                                   |                               |
|  |                              | <i>15</i>          | <i>17</i> | <i>32</i> | <i>1</i>   |                          |                            |                                   |                               |
| <b>1</b>   | <b>2</b>                     | <b>3</b>           | <b>4</b>  | <b>5</b>  | <b>6</b>   | <b>7</b>                 | <b>8</b>                   | <b>9</b>                          | <b>10</b>                     |
| <b>1.</b> Программное средство (ПС)                                      | 46                           | +                  | +         | +         | +          | 4                        | 11,5                       | Лекция, ЛР,<br>СРС                | зачет,<br>экзамен, кр         |
| <b>2.</b> Этапы разработки ПС  | 134                          | +                  | +         | +         | +          | 4                        | 33,5                       | Лекция, ЛР,<br>СРС                | зачет,<br>экзамен, кр         |
| <b>3.</b> Компьютерная поддержка разработки ПС                           | 31                           | +                  | +         | +         | +          | 4                        | 7,75                       | Лекция, ЛР,<br>СРС                | зачет,<br>экзамен, РГР        |
| <b>4.</b> Разработка интерфейса ПС                                       | 41                           | +                  | +         | +         | +          | 4                        | 10,25                      | Лекция, ЛР,<br>СРС                | зачет,<br>экзамен, РГР        |
| <i>всего часов</i>   | <b>252</b>                   | 63                 | 63        | 63        | 63         | <b>4</b>                 | <b>63</b>                  |                                   |                               |

## 6. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

1. Долинер, Л.И. Основы программирования в среде PascalABC.NET : учебное пособие / Л.И. Долинер; Министерство образования и науки Российской Федерации, Уральский федеральный университет имени первого Президента России Б. Н. Ельцина ; науч. ред. Г.А. Матвеева. - Екатеринбург : Издательство Уральского университета, 2014. - 129 с. : ил. - Библиогр. в кн. - ISBN 978-5-7996-1260-3 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=275988>.

## 7. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

| №                                | Наименование издания  | Вид занятия<br>(Лк, ЛР, кр, РГР) | Количество экземпляров в библиотеке, шт. | Обеспеченность, (экз./чел.) |
|----------------------------------|---|----------------------------------|--|-----------------------------|
| 1                                | 2   | 3                                | 4  | 5                           |
| <b>Основная литература</b>       |   |                                  |  |                             |
| 1.                               | Горохов, Д.Б. Программирование на языке Pascal: методические указания к выполнению лабораторных работ. - Братск: Изд-во БрГУ, 2017. - 144 с.  | Лк, ЛР, кр                       | 25                                       | 1                           |
| 2.                               | Горохов Д.Б. Программирование на языке Object Pascal : практикум. - Братск : Изд-во БрГУ, 2018. - 173 с.  | Лк, ЛР, кр                       | 25                                       | 1                           |
| 3.                               | Технология программирования / Ю.Ю. Громов, О.Г. Иванова, М.П. Беляев, Ю.В. Минин ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Тамбовский государственный технический университет». - Тамбов : Издательство ФГБОУ ВПО «ТГТУ», 2013. - 173 с. : ил. - Библиогр. в кн. - ISBN 978-5-8265-1207-4 ; То же [Электронный ресурс]. - URL: <a href="http://biblioclub.ru/index.php?page=book&amp;id=277802">http://biblioclub.ru/index.php?page=book&amp;id=277802</a> | Лк, ЛР, кр                       | ЭР                                       | 1                           |
| 4.                               | Ремонтов, А.П. Интерфейсы информационных систем : учебное пособие / А.П. Ремонтов, А.П. Писарев, Д.В. Строганов; Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Пензенский государственный технологический университет», Минобрнауки России. - Пенза : ПензГТУ, 2014. - 76 с. : схем., ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <a href="http://biblioclub.ru/index.php?page=book&amp;id=437155">http://biblioclub.ru/index.php?page=book&amp;id=437155</a>  | Лк, ЛР, РГР                      | ЭР                                       | 1                           |
| <b>Дополнительная литература</b> |   |                                  |  |                             |
| 5.                               | Зыков, С.В. Введение в теорию программирования. Объектно-ориентированный подход / С.В. Зыков. - 2-е изд., испр. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 189 с. : схем. - (Основы информационных технологий). - Библиогр. в кн. - ISBN 5-9556-0009-4 ; То же [Электронный ресурс]. - URL: <a href="http://biblioclub.ru/index.php?page=book&amp;id=429073">http://biblioclub.ru/index.php?page=book&amp;id=429073</a>   | Лк                               | ЭР                                       | 1                           |

|    |  |             |    |      |
|----|--|-------------|----|------|
| 6. | Непейвода, Н.Н. Стили и методы программирования / Н.Н. Непейвода. - М. : Интернет-Университет Информационных Технологий, 2005. - 320 с. - (Основы информационных технологий). - ISBN 5-9556-0023-X ; То же [Электронный ресурс]. - URL: <a href="http://biblioclub.ru/index.php?page=book&amp;id=233198">http://biblioclub.ru/index.php?page=book&amp;id=233198</a>  | Лк, ЛР, кр  | ЭР | 0,75 |
| 7. | Брокшмидт, К. Пользовательский интерфейс приложений для Windows 8, созданных с использованием HTML, CSS и JavaScript : учебный курс / К. Брокшмидт. - 2-е изд., исправ. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 396 с. : ил. ; То же [Электронный ресурс]. - URL: <a href="http://biblioclub.ru/index.php?page=book&amp;id=429247">http://biblioclub.ru/index.php?page=book&amp;id=429247</a> | Лк, ЛР, РГР | ЭР | 1    |

## 8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО - ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ» НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1. Электронный каталог библиотеки БрГУ  
[http://irbis.brstu.ru/CGI/irbis64r\\_15/cgiirbis\\_64.exe?LNG=&C21COM=F&I21DBN=BOOK&P21DBN=BOOK&S21CNR=&Z21ID=](http://irbis.brstu.ru/CGI/irbis64r_15/cgiirbis_64.exe?LNG=&C21COM=F&I21DBN=BOOK&P21DBN=BOOK&S21CNR=&Z21ID=)
2. Электронная библиотека БрГУ <http://ecat.brstu.ru/catalog> .
3. Электронно-библиотечная система «Университетская библиотека online»  
<http://biblioclub.ru> .
4. Электронно-библиотечная система «Издательство «Лань» <http://e.lanbook.com> .
5. Информационная система "Единое окно доступа к образовательным ресурсам"  
<http://window.edu.ru> .
6. Научная электронная библиотека eLIBRARY.RU <http://elibrary.ru> .
7. Университетская информационная система РОССИЯ (УИС РОССИЯ)  
<https://uisrussia.msu.ru/> .
8. Национальная электронная библиотека НЭБ <http://xn--90ax2c.xn--p1ai/how-to-search/> .

## 9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

| Вид учебных занятий                            | Организация деятельности обучающихся  |
|--|---|
| Лекции   | Написание конспекта лекций: кратко, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометить важные мысли, выделять ключевые слова, термины. Проверка терминов с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, практическом занятии. |
| Лабораторные работы, практические занятия, РГР | Работа с конспектом лекций, обобщение, систематизация, углубление и конкретизация полученных теоретических знаний, выработка способности и готовности их использования на практике. Развитие интеллектуальных умений, подготовка ответов к контрольным вопросам, работа с основной и дополнительной литературой, необходимой для освоения дисциплины, выполнение заданий, решение задач, активное участие в интерактивной, активной, инновационной формах обучения, составление письменных отчетов.   |
| Самостоятельная работа обучающихся             | <i>Подготовка к лабораторным работам.</i> Проработка основной и дополнительной литературы, терминов, сведений, формул требующихся для запоминания и являющихся основополагающими в разделе. Конспектирование прочитанных литера-  |

|  |  |
|--|--|
|  | <p>турных источников. Проработка материалов по изучаемому вопросу, с использованием на рекомендуемых ресурсах информационно-телекоммуникационной сети «Интернет». Выполнение заданий преподавателя, необходимых для подготовки к участию в интерактивной, активной, инновационных формах обучения по изучаемой теме.</p> <p><i>Подготовка к зачету, экзамену.</i> При подготовке к зачету, экзамену необходимо ориентироваться на конспекты лекций, рекомендуемую литературу, использовать рекомендуемые ресурсы информационно-телекоммуникационной сети «Интернет».</p> |
|--|--|

### 9.1. Методические указания для обучающихся по выполнению лабораторных работ

Для всех лабораторных:

Основная литература

Источники 1-3, указанные в разделе 7.

Дополнительная литература

Источники 5-6, указанные в разделе 7.

#### Лабораторная работа № 1. Целочисленные вычисления, CRT, GraphABC

Цель работы: получить первичные навыки работы в PascalABC.

Интерактив – работа в малых группах.

Задания:

Задание №1 по вариантам:

1. Дано расстояние  $L$  в сантиметрах. Используя операцию деления нацело, найти количество полных метров в нем (1 метр = 100 см).
2. Дана масса  $M$  в килограммах. Используя операцию деления нацело, найти количество полных тонн в ней (1 тонна = 1000 кг).
3. Дан размер файла в байтах. Используя операцию деления нацело, найти количество полных килобайтов, которые занимает данный файл (1 килобайт = 1024 байта).
4. Даны целые положительные числа  $A$  и  $B$  ( $A > B$ ). На отрезке длины  $A$  размещено максимально возможное количество отрезков длины  $B$  (без наложений). Используя операцию деления нацело, найти количество отрезков  $B$ , размещенных на отрезке  $A$ .
5. Даны целые положительные числа  $A$  и  $B$  ( $A > B$ ). На отрезке длины  $A$  размещено максимально возможное количество отрезков длины  $B$  (без наложений). Используя операцию взятия остатка от деления нацело, найти длину незанятой части отрезка  $A$ .
6. Дано двузначное число. Вывести вначале его левую цифру (десятки), а затем - его правую цифру (единицы). Для нахождения десятков использовать операцию деления нацело, для нахождения единиц - операцию взятия остатка от деления.
7. Дано двузначное число. Найти сумму и произведение его цифр.
8. Дано двузначное число. Вывести число, полученное при перестановке цифр исходного числа.
9. Дано трехзначное число. Используя одну операцию деления нацело, вывести первую цифру данного числа (сотни).
10. Дано трехзначное число. Вывести вначале его последнюю цифру (единицы), а затем - его среднюю цифру (десятки).
11. Дано трехзначное число. Найти сумму и произведение его цифр.
12. Дано трехзначное число. Вывести число, полученное при прочтении исходного числа справа налево.
13. Дано трехзначное число. В нем зачеркнули первую слева цифру и приписали ее справа. Вывести полученное число.
14. Дано трехзначное число. В нем зачеркнули первую справа цифру и приписали ее слева. Вывести полученное число.
15. Дано трехзначное число. Вывести число, полученное при перестановке цифр сотен и десятков исходного числа (например, 123 перейдет в 213).
16. Дано трехзначное число. Вывести число, полученное при перестановке цифр десятков и единиц исходного числа (например, 123 перейдет в 132).

17. Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду сотен в записи этого числа.
18. Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду тысяч в записи этого числа.
19. С начала суток прошло  $N$  секунд ( $N$  - целое). Найти количество полных минут, прошедших с начала суток.
20. С начала суток прошло  $N$  секунд ( $N$  - целое). Найти количество полных часов, прошедших с начала суток.
21. С начала суток прошло  $N$  секунд ( $N$  - целое). Найти количество секунд, прошедших с начала последней минуты.
22. С начала суток прошло  $N$  секунд ( $N$  - целое). Найти количество секунд, прошедших с начала последнего часа.
23. С начала суток прошло  $N$  секунд ( $N$  - целое). Найти количество полных минут, прошедших с начала последнего часа.
24. Дни недели пронумерованы следующим образом: 0 - воскресенье, 1 - понедельник, 2 - вторник, ..., 6 - суббота. Дано целое число  $K$ , лежащее в диапазоне 1-365. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было понедельником.
25. Дни недели пронумерованы следующим образом: 0 - воскресенье, 1 - понедельник, 2 - вторник, ..., 6 - суббота. Дано целое число  $K$ , лежащее в диапазоне 1-365. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было четвергом.
26. Дни недели пронумерованы следующим образом: 1 - понедельник, 2 - вторник, ..., 6 - суббота, 7 - воскресенье. Дано целое число  $K$ , лежащее в диапазоне 1-365. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было вторником.
27. Дни недели пронумерованы следующим образом: 1 - понедельник, 2 - вторник, ..., 6 - суббота, 7 - воскресенье. Дано целое число  $K$ , лежащее в диапазоне 1-365. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было субботой.
28. Дни недели пронумерованы следующим образом: 1 - понедельник, 2 - вторник, ..., 6 - суббота, 7 - воскресенье. Дано целое число  $K$ , лежащее в диапазоне 1-365, и целое число  $N$ , лежащее в диапазоне 1-7. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было днем недели с номером  $N$ .
29. Даны целые положительные числа  $A$ ,  $B$ ,  $C$ . На прямоугольнике размера  $A \times B$  размещено максимально возможное количество квадратов со стороной  $C$  (без наложений). Найти количество квадратов, размещенных на прямоугольнике, а также площадь незанятой части прямоугольника.
30. Дан номер некоторого года (целое положительное число). Определить соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.

Задание №2:

Напишите программу «Визитная карточка», которая выводила бы на экран хорошо оформленные ваши личные данные: имя, фамилию, год рождения, адрес, телефон. Используйте разные цвета, хорошо скомпонуйте.

Задание №3:

Напишите программу, использующую не менее трех процедур рисования, которая рисовала бы несложный рисунок (домик, автомобиль, и т.п.)

Порядок выполнения:

1. Запустить PascalABC.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе



При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

#### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 2. Управляющие конструкции**

Цель работы: получить навыки организации изменения потока команд в PascalABC.

Интерактив – работа в малых группах.

#### Задания:

Задание №1 по вариантам:

1. Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае не изменять его. Вывести полученное число.
  2. Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае вычесть из него 2. Вывести полученное число.
  3. Дано целое число. Если оно является положительным, то прибавить к нему 1; если отрицательным, то вычесть из него 2; если нулевым, то заменить его на 10. Вывести полученное число.
  4. Даны три целых числа. Найти количество положительных чисел в исходном наборе.
  5. Даны три целых числа. Найти количество положительных и количество отрицательных чисел в исходном наборе.
  6. Даны два числа. Вывести большее из них.
  7. Даны два числа. Вывести порядковый номер меньшего из них.
  8. Даны два числа. Вывести вначале большее, а затем меньшее из них.
  9. Даны две переменные вещественного типа:  $A$ ,  $B$ . Перераспределить значения данных переменных так, чтобы в  $A$  оказалось меньшее из значений, а в  $B$  - большее. Вывести новые значения переменных  $A$  и  $B$ .
  10. Даны две переменные целого типа:  $A$  и  $B$ . Если их значения не равны, то присвоить каждой переменной сумму этих значений, а если равны, то присвоить переменным нулевые значения. Вывести новые значения переменных  $A$  и  $B$ .
  11. Даны две переменные целого типа:  $A$  и  $B$ . Если их значения не равны, то присвоить каждой переменной большее из этих значений, а если равны, то присвоить переменным нулевые значения. Вывести новые значения переменных  $A$  и  $B$ .
  12. Даны три числа. Найти наименьшее из них.
  13. Даны три числа. Найти среднее из них (то есть число, расположенное между наименьшим и наибольшим).
  14. Даны три числа. Вывести вначале наименьшее, а затем наибольшее из данных чисел.
  15. Даны три числа. Найти сумму двух наибольших из них.
  16. Даны три переменные вещественного типа:  $A$ ,  $B$ ,  $C$ . Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных  $A$ ,  $B$ ,  $C$ .
  17. Даны три переменные вещественного типа:  $A$ ,  $B$ ,  $C$ . Если их значения упорядочены по возрастанию или убыванию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных  $A$ ,  $B$ ,  $C$ .
  18. Даны три целых числа, одно из которых отлично от двух других, равных между собой. Определить порядковый номер числа, отличного от остальных.
  19. Даны четыре целых числа, одно из которых отлично от трех других, равных между собой. Определить порядковый номер числа, отличного от остальных.
  20. На числовой оси расположены три точки:  $A$ ,  $B$ ,  $C$ . Определить, какая из двух последних точек ( $B$  или  $C$ ) расположена ближе к  $A$ , и вывести эту точку и ее расстояние от точки  $A$ .
- Задание №2 по вариантам (оператор Case):
1. Дано целое число в диапазоне 1-7. Вывести строку - название дня недели, соответствующее данному числу (1 - «понедельник», 2 - «вторник» и т. д.).

2. Дано целое число  $K$ . Вывести строку-описание оценки, соответствующей числу  $K$  (1 - «плохо», 2 - «неудовлетворительно», 3 - «удовлетворительно», 4 - «хорошо», 5 - «отлично»). Если  $K \notin [1,5]$ , то вывести строку «ошибка».
3. Дан номер месяца - целое число в диапазоне 1-12 (1 - январь, 2 - февраль и т. д.). Вывести название соответствующего времени года («зима», «весна», «лето», «осень»).
4. Дан номер месяца - целое число в диапазоне 1-12 (1 - январь, 2 - февраль и т. д.). Определить количество дней в этом месяце для невисокосного года.
5. Арифметические действия над числами пронумерованы следующим образом: 1 - сложение, 2 - вычитание, 3 - умножение, 4 - деление. Дан номер действия  $N$  (целое число в диапазоне 1-4) и вещественные числа  $A$  и  $B$  ( $B$  не равно 0). Выполнить над числами указанное действие и вывести результат.
6. Единицы длины пронумерованы следующим образом: 1 - дециметр, 2 - километр, 3 - метр, 4 - миллиметр, 5 - сантиметр. Дан номер единицы длины (целое число в диапазоне 1-5) и длина отрезка в этих единицах (вещественное число). Найти длину отрезка в метрах.
7. Единицы массы пронумерованы следующим образом: 1 - килограмм, 2 - миллиграмм, 3 - грамм, 4 - тонна, 5 - центнер. Дан номер единицы массы (целое число в диапазоне 1-5) и масса тела в этих единицах (вещественное число). Найти массу тела в килограммах.
8. Даны два целых числа:  $D$  (день) и  $M$  (месяц), определяющие правильную дату невисокосного года. Вывести значения  $D$  и  $M$  для даты, предшествующей указанной.
9. Даны два целых числа:  $D$  (день) и  $M$  (месяц), определяющие правильную дату невисокосного года. Вывести значения  $D$  и  $M$  для даты, следующей за указанной.
10. Робот может перемещаться в четырех направлениях («С» - север, «З» - запад, «Ю» - юг, «В» - восток) и принимать три цифровые команды: 0 - продолжать движение, 1 - поворот налево, (-1) - поворот направо. Дан символ  $C$  - исходное направление робота и целое число  $N$  - посланная ему команда. Вывести направление робота после выполнения полученной команды.
11. Локатор ориентирован на одну из сторон света («С» - север, «З» - запад, «Ю» - юг, «В» - восток) и может принимать три цифровые команды поворота: 1 - поворот налево, -1 - поворот направо, 2 - поворот на  $180^\circ$ . Дан символ  $C$  - исходная ориентация локатора и целые числа  $N1$  и  $N2$  - две посланные команды. Вывести ориентацию локатора после выполнения этих команд.
12. Элементы окружности пронумерованы следующим образом: 1 - радиус  $R$ , 2 - диаметр  $D = 2R$ , 3 - длина  $L = 2\pi R$ , 4 - площадь круга  $S = \pi R^2$ . Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данной окружности (в том же порядке).
13. Элементы равнобедренного прямоугольного треугольника пронумерованы следующим образом: 1 - катет  $a$ , 2 - гипотенуза  $c = \sqrt{2}a$ , 3 - высота  $h$ , опущенная на гипотенузу ( $h = c/2$ ), 4 - площадь  $S = ch/2$ . Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).
14. Элементы равностороннего треугольника пронумерованы следующим образом: 1 - сторона  $a$ , 2 - радиус  $R_1$  вписанной окружности ( $R_1 = a\sqrt{3}/6$ ), 3 - радиус  $R_2$  описанной окружности ( $R_2 = 2R_1$ ), 4 - площадь  $S = a^2\sqrt{3}/4$ . Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).
15. Мастям игральных карт присвоены порядковые номера: 1 - пики, 2 - трефы, 3 - бубны, 4 - червы. Достоинству карт, старших десятки, присвоены номера: 11 - валет, 12 - дама, 13 - король, 14 - туз. Даны два целых числа:  $N$  - достоинство ( $6 < N < 14$ ) и  $M$  - масть карты ( $1 < M < 4$ ). Вывести название соответствующей карты вида «шестерка бубен», «дама червей», «туз треф» и т. п.
16. Дано целое число в диапазоне 20-69, определяющее возраст (в годах). Вывести строку-описание указанного возраста, обеспечив правильное согласование числа со словом «год», например: 20 - «двадцать лет», 32 - «тридцать два года», 41 - «сорок один год».
17. Дано целое число в диапазоне 10-40, определяющее количество учебных заданий по некоторой теме. Вывести строку-описание указанного количества заданий, обеспечив правильное согласование числа со словами «учебное задание», например: 18 - «восемнадцать учебных заданий», 23 - «двадцать три учебных задания», 31 - «тридцать одно учебное задание».

18. Дано целое число в диапазоне 100-999. Вывести строку-описание данного числа, например: 256 - «двести пятьдесят шесть».

19. В восточном календаре принят 60-летний цикл, состоящий из 12-летних подциклов, обозначаемых названиями цвета: зеленый, красный, желтый, белый и черный. В каждом подцикле годы носят названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи. По номеру года определить его название, если 1984 год - начало цикла: «год зеленой крысы».

20. Даны два целых числа:  $D$  (день) и  $M$  (месяц), определяющие правильную дату. Вывести знак Зодиака, соответствующий этой дате: «Водолей» (20.1-18.2), «Рыбы» (19.2-20.3), «Овен» (21.3-19.4), «Телец» (20.4-20.5), «Близнецы» (21.5-21.6), «Рак» (22.6-22.7), «Лев» (23.7-22.8), «Дева» (23.8-22.9), «Весы» (23.9-22.10), «Скорпион» (23.10-22.11), «Стрелец» (23.11-21.12), «Козерог» (22.12-19.1).

Задание №3 по вариантам (оператор For):

1. Даны целые числа  $K$  и  $N$  ( $N > 0$ ). Вывести  $N$  раз число  $K$ .

2. Даны два целых числа  $A$  и  $B$  ( $A < B$ ). Вывести в порядке возрастания все целые числа, расположенные между  $A$  и  $B$  (включая сами числа  $A$  и  $B$ ), а также количество  $N$  этих чисел.

3. Даны два целых числа  $A$  и  $B$  ( $A < B$ ). Вывести в порядке убывания все целые числа, расположенные между  $A$  и  $B$  (не включая числа  $A$  и  $B$ ), а также количество  $N$  этих чисел.

4. Дано вещественное число - цена 1 кг конфет. Вывести стоимость 1, 2, ..., 10 кг конфет.

5. Дано вещественное число - цена 1 кг конфет. Вывести стоимость 0.1, 0.2, ..., 1 кг конфет.

6. Дано вещественное число - цена 1 кг конфет. Вывести стоимость 1.2, 1.4, ..., 2 кг конфет.

7. Даны два целых числа  $A$  и  $B$  ( $A < B$ ). Найти сумму всех целых чисел от  $A$  до  $B$  включительно.

8. Даны два целых числа  $A$  и  $B$  ( $A < B$ ). Найти произведение всех целых чисел от  $A$  до  $B$  включительно.

9. Даны два целых числа  $A$  и  $B$  ( $A < B$ ). Найти сумму квадратов всех целых чисел от  $A$  до  $B$  включительно.

10. Дано целое число  $N > 0$ . Найти сумму  $1 + 1/2 + 1/3 + \dots + 1/N$ .

11. Дано целое число  $N > 0$ . Найти сумму  $N^2 + (N+1)^2 + (N+2)^2 + \dots + (2N)^2$ .

12. Дано целое число  $N > 0$ . Найти произведение  $1, 1 \cdot 1, 2 \cdot 1, 3 \cdot \dots$  ( $N$  сомножителей).

13. Дано целое число  $N > 0$ . Найти значение выражения  $1, 1-1, 2+1, 3-\dots$  ( $N$  слагаемых, знаки чередуются). Условный оператор не использовать.

14. Дано целое число  $N > 0$ . Найти квадрат данного числа, используя для его вычисления следующую формулу:  $N^2 = 1 + 3 + 5 + \dots + (2N-1)$ . После добавления к сумме каждого слагаемого выводить текущее значение суммы (в результате будут выведены квадраты всех целых чисел от 1 до  $N$ ).

15. Дано вещественное число  $A$  и целое число  $N$  ( $> 0$ ). Найти  $A$  в степени  $N$ :  $A^N = A \cdot A \dots \cdot A$  (числа  $A$  перемножаются  $N$  раз).

16. Дано вещественное число  $A$  и целое число  $N$  ( $> 0$ ). Используя один цикл, вывести все целые степени числа  $A$  от 1 до  $N$ .

17. Дано вещественное число  $A$  и целое число  $N$  ( $> 0$ ). Используя один цикл, найти сумму  $1 + A + A^2 + A^3 + \dots + A^N$ .

18. Дано вещественное число  $A$  и целое число  $N$  ( $> 0$ ). Используя один цикл, найти значение выражения  $1 - A + A^2 - A^3 + \dots + (-1)^N A^N$ . Оператор if не использовать.

19. Дано целое число  $N > 0$ . Найти произведение  $N! = 1 \cdot 2 \cdot \dots \cdot N$  ( $N$ -факториал). Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и вывести его как вещественное число.

20. Дано целое число  $N > 0$ . Используя один цикл, найти сумму  $1! + 2! + 3! + \dots + N!$  (выражение  $N!$  -  $N$ -факториал - обозначает произведение всех целых чисел от 1 до  $N$ :  $N! = 1 \cdot 2 \cdot \dots \cdot N$ ). Чтобы избежать целочисленного переполнения, проводить вычисления с помощью вещественных переменных и вывести результат как вещественное число.

Задание №4 по вариантам (оператор While):

1. Даны положительные числа  $A$  и  $B$  ( $A > B$ ). На отрезке длины  $A$  размещено максимально возможное количество отрезков длины  $B$  (без наложений). Не используя операции умножения и деления, найти длину незанятой части отрезка  $A$ .

2. Даны положительные числа  $A$  и  $B$  ( $A > B$ ). На отрезке длины  $A$  размещено максимально возможное количество отрезков длины  $B$  (без наложений). Не используя операции умножения и деления, найти количество отрезков  $B$ , размещенных на отрезке  $A$ .
3. Даны целые положительные числа  $N$  и  $K$ . Используя только операции сложения и вычитания, найти частное от деления нацело  $N$  на  $K$ , а также остаток от этого деления.
4. Дано целое число  $N > 0$ . Если оно является степенью числа 3, то вывести TRUE, если не является - вывести FALSE.
5. Дано целое число  $N > 0$ , являющееся некоторой степенью числа 2:  $N = 2^K$ . Найти целое число  $K$  - показатель этой степени.
6. Дано целое число  $N > 0$ . Найти *двойной факториал*  $N$ :  $N!! = N \cdot (N-2) \cdot (N-4) \cdot \dots$  (последний сомножитель равен 2, если  $N$  - четное, и 1, если  $N$  - нечетное). Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и вывести его как вещественное число.
7. Дано целое число  $N > 0$ . Найти наименьшее целое положительное число  $K$ , квадрат которого превосходит  $N$ :  $K^2 > N$ . Функцию извлечения квадратного корня не использовать.
8. Дано целое число  $N > 0$ . Найти наибольшее целое число  $K$ , квадрат которого не превосходит  $N$ :  $K^2 < N$ . Функцию извлечения квадратного корня не использовать.
9. Дано целое число  $N > 0$ . Найти наименьшее целое число  $K$ , при котором выполняется неравенство  $3K > N$ .
10. Дано целое число  $N > 0$ . Найти наибольшее целое число  $K$ , при котором выполняется неравенство  $3K < N$ .
11. Дано целое число  $N > 0$ . Вывести наименьшее из целых чисел  $K$ , для которых сумма  $1+2+\dots+K$  будет больше или равна  $N$ , и саму эту сумму.
12. Дано целое число  $N > 1$ . Вывести наибольшее из целых чисел  $K$ , для которых сумма  $1+2+\dots+K$  будет меньше или равна  $N$ , и саму эту сумму.
13. Дано число  $A > 1$ . Вывести наименьшее из целых чисел  $K$ , для которых сумма  $1+1/2+\dots+1/K$  будет больше  $A$ , и саму эту сумму.
14. Дано число  $A > 1$ . Вывести наибольшее из целых чисел  $K$ , для которых сумма  $1+1/2+\dots+1/K$  будет меньше  $A$ , и саму эту сумму.
15. Начальный вклад в банке равен 1000 руб. Через каждый месяц размер вклада увеличивается на  $P$  процентов от имеющейся суммы ( $P$  - вещественное число,  $0 < P < 25$ ). По данному  $P$  определить, через сколько месяцев размер вклада превысит 1100 руб., и вывести найденное количество месяцев  $K$  (целое число) и итоговый размер вклада  $S$  (вещественное число).
16. Спортсмен-лыжник начал тренировки, пробежав в первый день 10 км. Каждый следующий день он увеличивал длину пробега на  $P$  процентов от пробега предыдущего дня ( $P$  - вещественное,  $0 < P < 50$ ). По данному  $P$  определить, после какого дня суммарный пробег лыжника за все дни превысит 200 км, и вывести найденное количество дней  $K$  (целое) и суммарный пробег  $S$  (вещественное число).
17. Дано целое число  $N > 0$ . Используя операции деления нацело и взятия остатка от деления, вывести все его цифры, начиная с самой правой (разряда единиц).
18. Дано целое число  $N > 0$ . Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.
19. Дано целое число  $N > 0$ . Используя операции деления нацело и взятия остатка от деления, найти число, полученное при прочтении числа  $N$  справа налево.
20. Дано целое число  $N > 0$ . С помощью операций деления нацело и взятия остатка от деления определить, имеется ли в записи числа  $N$  цифра «2». Если имеется, то вывести TRUE, если нет - вывести FALSE.

Порядок выполнения:

1. Запустить PascalABC.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.

2. Провести отладку и тестирование.

### Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

## **Лабораторная работа № 3. Подпрограммы и функции**

Цель работы: получить навыки процедурного программирования в PascalABC.

Интерактив – работа в малых группах.

### Задания:

Задание №1 по вариантам (Функции с числовыми параметрами):

1. Описать функцию  $Sign(X)$  целого типа, возвращающую для вещественного числа  $X$  следующие значения:  $(-1)$ , если  $X < 0$ ;  $0$ , если  $X = 0$ ;  $1$ , если  $X > 0$ . С помощью этой функции найти значение выражения  $Sign(A) + Sign(B)$  для данных вещественных чисел  $A$  и  $B$ .
2. Описать функцию  $RootsCount(A, B, C)$  целого типа, определяющую количество корней квадратного уравнения  $Ax^2 + Bx + C = 0$  ( $A, B, C$  - вещественные параметры,  $A \neq 0$ ). С ее помощью найти количество корней для каждого из трех квадратных уравнений с данными коэффициентами. Количество корней определять по значению дискриминанта:  $D = B^2 - 4AC$ .
3. Описать функцию  $CircleS(R)$  вещественного типа, находящую площадь круга радиуса  $R$  ( $R$  - вещественное). С помощью этой функции найти площади трех кругов с данными радиусами. Площадь круга радиуса  $R$  вычисляется по формуле  $S = \pi R^2$ .
4. Описать функцию  $RingS(R_1, R_2)$  вещественного типа, находящую площадь кольца, заключенного между двумя окружностями с общим центром и радиусами  $R_1$  и  $R_2$  ( $R_1$  и  $R_2$  - вещественные,  $R_1 > R_2$ ). С ее помощью найти площади трех колец, для которых даны внешние и внутренние радиусы. Воспользоваться формулой площади круга радиуса  $R$ :  $S = \pi R^2$ .
5. Описать функцию  $TriangleP(a, h)$ , находящую периметр равнобедренного треугольника по его основанию  $a$  и высоте  $h$ , проведенной к основанию ( $a$  и  $h$  - вещественные). С помощью этой функции найти периметры трех треугольников, для которых даны основания и высоты. Для нахождения боковой стороны  $b$  треугольника использовать теорему Пифагора:  $b^2 = (a/2)^2 + h^2$ .
6. Описать функцию  $SumRange(A, B)$  целого типа, находящую сумму всех целых чисел от  $A$  до  $B$  включительно ( $A$  и  $B$  - целые). Если  $A > B$ , то функция возвращает  $0$ . С помощью этой функции найти суммы чисел от  $A$  до  $B$  и от  $B$  до  $C$ , если даны числа  $A, B, C$ .
7. Описать функцию  $Calc(A, B, Op)$  вещественного типа, выполняющую над ненулевыми вещественными числами  $A$  и  $B$  одну из арифметических операций и возвращающую ее результат. Вид операции определяется целым параметром  $Op$ :  $1$  - вычитание,  $2$  - умножение,  $3$  - деление, остальные значения - сложение. С помощью  $Calc$  выполнить для данных  $A$  и  $B$  операции, определяемые данными целыми  $N_1, N_2, N_3$ .
8. Описать функцию  $Quarter(x, y)$  целого типа, определяющую номер координатной четверти, в которой находится точка с ненулевыми вещественными координатами  $(x, y)$ . С помощью этой функции найти номера координатных четвертей для трех точек с данными ненулевыми координатами.
9. Описать функцию  $Even(K)$  логического типа, возвращающую TRUE, если целый параметр  $K$  является четным, и FALSE в противном случае. С ее помощью найти количество четных чисел в наборе из  $10$  целых чисел.
10. Описать функцию  $IsSquare(K)$  логического типа, возвращающую TRUE, если целый параметр  $K (>0)$  является квадратом некоторого целого числа, и FALSE в противном случае. С ее помощью найти количество квадратов в наборе из  $10$  целых положительных чисел.

11. Описать функцию  $IsPower5(K)$  логического типа, возвращающую TRUE, если целый параметр  $K (> 0)$  является степенью числа 5, и FALSE в противном случае. С ее помощью найти количество степеней числа 5 в наборе из 10 целых положительных чисел.
12. Описать функцию  $IsPowerN(K, N)$  логического типа, возвращающую TRUE, если целый параметр  $K (> 0)$  является степенью числа  $N (> 1)$ , и FALSE в противном случае. Дано число  $N (> 1)$  и набор из 10 целых положительных чисел. С помощью функции  $IsPowerN$  найти количество степеней числа  $N$  в данном наборе.
13. Описать функцию  $IsPrime(N)$  логического типа, возвращающую TRUE, если целый параметр  $N (> 1)$  является простым числом, и FALSE в противном случае (число, большее 1, называется простым, если оно не имеет положительных делителей, кроме 1 и самого себя). Дан набор из 10 целых чисел, больших 1. С помощью функции  $IsPrime$  найти количество простых чисел в данном наборе.
14. Описать функцию  $DigitCount(K)$  целого типа, находящую количество цифр целого положительного числа  $K$ . Используя эту функцию, найти количество цифр для каждого из пяти данных целых положительных чисел.
15. Описать функцию  $DigitN(K, N)$  целого типа, возвращающую  $N$ -ю цифру целого положительного числа  $K$  (цифры в числе нумеруются справа налево). Если количество цифр в числе  $K$  меньше  $N$ , то функция возвращает - 1. Для каждого из пяти данных целых положительных чисел  $K_1, K_2, \dots, K_5$  вызвать функцию  $DigitN$  с параметром  $N$ , изменяющимся от 1 до 5.
16. Описать функцию  $IsPalindrom(K)$ , возвращающую TRUE, если целый параметр  $K (> 0)$  является палиндромом (то есть его запись читается одинаково слева направо и справа налево), и FALSE в противном случае. С ее помощью найти количество палиндромов в наборе из 10 целых положительных чисел.
17. Описать функцию  $DegToRad(D)$  вещественного типа, находящую величину угла в радианах, если дана его величина  $D$  в градусах ( $D$  - вещественное число,  $0 < D < 360$ ). Воспользоваться следующим соотношением:  $180^\circ = \pi$  радианов. С помощью функции  $DegToRad$  перевести из градусов в радианы пять данных углов.
18. Описать функцию  $RadToDeg(R)$  вещественного типа, находящую величину угла в градусах, если дана его величина  $R$  в радианах ( $R$  - вещественное число,  $0 < R < 2\pi$ ). Воспользоваться следующим соотношением:  $180^\circ = \pi$  радианов. С помощью функции  $RadToDeg$  перевести из радианов в градусы пять данных углов.
19. Описать функцию  $Fact(N)$  вещественного типа, вычисляющую значение факториала  $N! = 12 \dots N$  ( $N > 0$  - параметр целого типа; вещественное возвращаемое значение используется для того, чтобы избежать целочисленного переполнения при больших значениях  $N$ ). С помощью этой функции найти факториалы пяти данных целых чисел.
20. Описать функцию  $Fact2(N)$  вещественного типа, вычисляющую двойной факториал:  
Задание №2 по вариантам (Процедуры с числовыми параметрами):
  1. Описать процедуру  $PowerA3(A, B)$ , вычисляющую третью степень числа  $A$  и возвращающую ее в переменной  $B$  ( $A$  - входной,  $B$  - выходной параметр; оба параметра являются вещественными). С помощью этой процедуры найти третьи степени пяти данных чисел.
  2. Описать процедуру  $PowerA234(A, B, C, D)$ , вычисляющую вторую, третью и четвертую степень числа  $A$  и возвращающую эти степени соответственно в переменных  $B, C$  и  $D$  ( $A$  - входной,  $B, C, D$  - выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти вторую, третью и четвертую степени пяти данных чисел.
  3. Описать процедуру  $Mean(X, Y, AMean, GMean)$ , вычисляющую среднее арифметическое  $AMean = (X+Y)/2$  и среднее геометрическое  $GMEAN = \sqrt{XY}$  двух положительных чисел  $X$  и  $Y$  ( $X$  и  $Y$  - входные,  $AMean$  и  $GMean$  - выходные параметры вещественного типа). С помощью этой процедуры найти среднее арифметическое и среднее геометрическое для пар  $(A, B)$ ,  $(A, C)$ ,  $(A, D)$ , если даны  $A, B, C, D$ .
  4. Описать процедуру  $TrianglePS(a, P, S)$ , вычисляющую по стороне  $a$  равностороннего треугольника его периметр  $P = 3a$  и площадь  $S = a^2 \sqrt{3}/4$  ( $a$  - входной,  $P$  и  $S$  - выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти периметры и площади трех равносторонних треугольников с данными сторонами.

5. Описать процедуру  $RectPS(x_1, y_1, x_2, y_2, P, S)$ , вычисляющую периметр  $P$  и площадь  $S$  прямоугольника со сторонами, параллельными осям координат, по координатам  $(x_1, y_1)$ ,  $(x_2, y_2)$  его противоположных вершин ( $x_1, y_1, x_2, y_2$  - входные,  $P$  и  $S$  - выходные параметры вещественного типа). С помощью этой процедуры найти периметры и площади трех прямоугольников с данными противоположными вершинами.
6. Описать процедуру  $DigitCountSum(K, C, S)$ , находящую количество  $C$  цифр целого положительного числа  $K$ , а также их сумму  $S$  ( $K$  - входной,  $C$  и  $S$  - выходные параметры целого типа). С помощью этой процедуры найти количество и сумму цифр для каждого из пяти данных целых чисел.
7. Описать процедуру  $InvertDigits(K)$ , меняющую порядок следования цифр целого положительного числа  $K$  на обратный ( $K$  - параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры поменять порядок следования цифр на обратный для каждого из пяти данных целых чисел.
8. Описать процедуру  $AddRightDigit(D, K)$ , добавляющую к целому положительному числу  $K$  справа цифру  $D$  ( $D$  - входной параметр целого типа, лежащий в диапазоне 0-9,  $K$  - параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры последовательно добавить к данному числу  $K$  справа данные цифры  $D1$  и  $D2$ , выводя результат каждого добавления.
9. Описать процедуру  $AddLeftDigit(D, K)$ , добавляющую к целому положительному числу  $K$  слева цифру  $D$  ( $D$  - входной параметр целого типа, лежащий в диапазоне 1-9,  $K$  - параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры последовательно добавить к данному числу  $K$  слева данные цифры  $D1$  и  $D2$ , выводя результат каждого добавления.
10. Описать процедуру  $Swap(X, Y)$ , меняющую содержимое переменных  $X$  и  $Y$  ( $X$  и  $Y$  - вещественные параметры, являющиеся одновременно входными и выходными). С ее помощью для данных переменных  $A, B, C, D$  последовательно поменять содержимое следующих пар:  $A$  и  $B$ ,  $C$  и  $D$ ,  $B$  и  $C$  и вывести новые значения  $A, B, C, D$ .
11. Описать процедуру  $Minmax(X, Y)$ , записывающую в переменную  $X$  минимальное из значений  $X$  и  $Y$ , а в переменную  $Y$  - максимальное из этих значений ( $X$  и  $Y$  - вещественные параметры, являющиеся одновременно входными и выходными). Используя четыре вызова этой процедуры, найти минимальное и максимальное из данных чисел  $A, B, C, D$ .
12. Описать процедуру  $SortInc3(A, B, C)$ , меняющую содержимое переменных  $A, B, C$  таким образом, чтобы их значения оказались упорядоченными по возрастанию ( $A, B, C$  - вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры упорядочить по возрастанию два данных набора из трех чисел:  $(A_1, B_1, C_1)$  и  $(A_2, B_2, C_2)$ .
13. Описать процедуру  $SortDec3(A, B, C)$ , меняющую содержимое переменных  $A, B, C$  таким образом, чтобы их значения оказались упорядоченными по убыванию ( $A, B, C$  - вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры упорядочить по убыванию два данных набора из трех чисел:  $(A_1, B_1, C_1)$  и  $(A_2, B_2, C_2)$ .
14. Описать процедуру  $ShiftRight3(A, B, C)$ , выполняющую правый циклический сдвиг: значение  $A$  переходит в  $B$ , значение  $B$  - в  $C$ , значение  $C$  в  $A$  ( $A, B, C$  - вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры выполнить правый циклический сдвиг для двух данных наборов из трех чисел:  $(A_1, B_1, C_1)$  и  $(A_2, B_2, C_2)$ .
15. Описать процедуру  $ShiftLeft3(A, B, C)$ , выполняющую левый циклический сдвиг: значение  $A$  переходит в  $C$ , значение  $C$  - в  $B$ , значение  $B$  в  $A$  ( $A, B, C$  - вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры выполнить левый циклический сдвиг для двух данных наборов из трех чисел:  $(A_1, B_1, C_1)$  и  $(A_2, B_2, C_2)$ .

Задание №3:

Сформируйте модуль с процедурами и функциями из предыдущих заданий.

Порядок выполнения:

1. Запустить PascalABC.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

#### **Лабораторная работа № 4. Массивы**

Цель работы: получить навыки работы с массивами в PascalABC.

Интерактив – работа в малых группах.

Задания:

Задание №1 по вариантам (одномерные массивы):

1. Дан массив  $A$  ненулевых целых чисел размера 10. Вывести значение первого из тех его элементов  $A_K$ , которые удовлетворяют неравенству  $A_K < A_{10}$ . Если таких элементов нет, то вывести 0.
2. Дан целочисленный массив  $A$  размера 10. Вывести порядковый номер последнего из тех его элементов  $A_K$ , которые удовлетворяют двойному неравенству  $A_1 < A_K < A_{10}$ . Если таких элементов нет, то вывести 0.
3. Дан массив размера  $N$  и целые числа  $K$  и  $L$  ( $1 < K < L < N$ ). Найти сумму элементов массива с номерами от  $K$  до  $L$  включительно.
4. Дан массив размера  $N$  и целые числа  $K$  и  $L$  ( $1 < K < L < N$ ). Найти среднее арифметическое элементов массива с номерами от  $K$  до  $L$  включительно.
5. Дан массив размера  $N$  и целые числа  $K$  и  $L$  ( $1 < K < L < N$ ). Найти сумму всех элементов массива, кроме элементов с номерами от  $K$  до  $L$  включительно.
6. Дан массив размера  $N$  и целые числа  $K$  и  $L$  ( $1 < K < L < N$ ). Найти среднее арифметическое всех элементов массива, кроме элементов с номерами от  $K$  до  $L$  включительно.
7. Дан целочисленный массив размера  $N$ , не содержащий одинаковых чисел. Проверить, образуют ли его элементы *арифметическую прогрессию*. Если образуют, то вывести разность прогрессии, если нет - вывести 0.
8. Дан массив ненулевых целых чисел размера  $N$ . Проверить, образуют ли его элементы *геометрическую прогрессию*. Если образуют, то вывести знаменатель прогрессии, если нет - вывести 0.
9. Дан целочисленный массив размера  $N$ . Проверить, чередуются ли в нем четные и нечетные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.
10. Дан массив ненулевых целых чисел размера  $N$ . Проверить, чередуются ли в нем положительные и отрицательные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.
11. Дан массив  $A$  размера  $N$ . Найти минимальный элемент из его элементов с четными номерами:  $A_2, A_4, A_6, \dots$ .
12. Дан массив  $A$  размера  $N$ . Найти максимальный элемент из его элементов с нечетными номерами:  $A_1, A_3, A_5, \dots$ .
13. Дан массив размера  $N$ . Найти номера тех элементов массива, которые больше своего правого соседа, и количество таких элементов. Найденные номера выводить в порядке их возрастания.
14. Дан массив размера  $N$ . Найти номера тех элементов массива, которые больше своего левого соседа, и количество таких элементов. Найденные номера выводить в порядке их убывания.



15. Дан массив размера  $N$ . Найти номер его первого локального минимума (*локальный минимум* - это элемент, который меньше любого из своих соседей).
16. Дан массив размера  $N$ . Найти номер его последнего локального максимума (*локальный максимум* - это элемент, который больше любого из своих соседей).
17. Дан массив размера  $N$ . Найти максимальный из его локальных минимумов (*локальный минимум* - это элемент, который меньше любого из своих соседей).
18. Дан массив размера  $N$ . Найти минимальный из его локальных максимумов (*локальный максимум* - это элемент, который больше любого из своих соседей).
19. Дан массив размера  $N$ . Найти максимальный из его элементов, не являющихся ни локальным минимумом, ни локальным максимумом (*локальный минимум* - это элемент, который меньше любого из своих соседей, *локальный максимум* - это элемент, который больше любого из своих соседей). Если таких элементов в массиве нет, то вывести 0 (как вещественное число).
20. Дан массив размера  $N$ . Найти количество участков, на которых его элементы возрастают. Задание №2 по вариантам (одномерные массивы):
  1. Даны массивы  $A$  и  $B$  одинакового размера  $N$ . Поменять местами их содержимое и вывести вначале элементы преобразованного массива  $A$ , а затем - элементы преобразованного массива  $B$ .
  2. Дан массив  $A$  размера  $N$ . Сформировать новый массив  $B$  того же размера, элементы которого определяются следующим образом:  $B_k = 2 - A_k$ , если  $A_k < 5$ ,  $A_k/2$  в противном случае.
  3. Даны два массива  $A$  и  $B$  одинакового размера  $N$ . Сформировать новый массив  $C$  того же размера, каждый элемент которого равен максимальному из элементов массивов  $A$  и  $B$  с тем же индексом.
  4. Дан целочисленный массив  $A$  размера  $N$ . Переписать в новый целочисленный массив  $B$  все четные числа из исходного массива (в том же порядке) и вывести размер полученного массива  $B$  и его содержимое.
  5. Дан целочисленный массив  $A$  размера  $N$  ( $< 15$ ). Переписать в новый целочисленный массив  $B$  все элементы с нечетными порядковыми номерами (1, 3, ...) и вывести размер полученного массива  $B$  и его содержимое. Условный оператор не использовать.
  6. Дан целочисленный массив  $A$  размера  $N$  ( $< 15$ ). Переписать в новый целочисленный массив  $B$  все элементы с порядковыми номерами, кратными трем (3, 6, ...), и вывести размер полученного массива  $B$  и его содержимое. Условный оператор не использовать.
  7. Дан целочисленный массив  $A$  размера  $N$ . Переписать в новый целочисленный массив  $B$  того же размера вначале все элементы исходного массива с четными номерами, а затем - с нечетными:  $A_2, A_4, A_6, \dots, A_1, A_3, A_5, \dots$ . Условный оператор не использовать.
  8. Дан массив  $A$  размера  $N$ . Сформировать новый массив  $B$  того же размера по следующему правилу: элемент  $B_K$  равен сумме элементов массива  $A$  с номерами от 1 до  $K$ .
  9. Дан массив  $A$  размера  $N$ . Сформировать новый массив  $B$  того же размера по следующему правилу: элемент  $B_K$  равен среднему арифметическому элементов массива  $A$  с номерами от 1 до  $K$ .
  10. Дан массив  $A$  размера  $N$ . Сформировать новый массив  $B$  того же размера по следующему правилу: элемент  $B_K$  равен сумме элементов массива  $A$  с номерами от  $K$  до  $N$ .
  11. Дан массив  $A$  размера  $N$ . Сформировать новый массив  $B$  того же размера по следующему правилу: элемент  $B_K$  равен среднему арифметическому элементов массива  $A$  с номерами от  $K$  до  $N$ .
  12. Дан массив  $A$  размера  $N$ . Сформировать два новых массива  $B$  и  $C$ : в массив  $B$  записать все положительные элементы массива  $A$ , в массив  $C$  - все отрицательные (сохраняя исходный порядок следования элементов). Вывести вначале размер и содержимое массива  $B$ , а затем - размер и содержимое массива  $C$ .
  13. Даны два массива  $A$  и  $B$  размера 5, элементы которых упорядочены по возрастанию. Объединить эти массивы так, чтобы результирующий массив  $C$  (размера 10) остался упорядоченным по возрастанию.
  14. Даны три целочисленных массива  $A$ ,  $B$  и  $C$  размера  $N_A$ ,  $N_B$ ,  $N_C$  соответственно, элементы которых упорядочены по убыванию. Объединить эти массивы так, чтобы результирующий целочисленный массив  $D$  (размера  $N_A + N_B + N_C$ ) остался упорядоченным по убыванию.

Задание №3 по вариантам (одномерные массивы):

1. Дан массив  $A$  размера  $N (< 6)$ . Упорядочить его по возрастанию методом сортировки *простым выбором*: найти максимальный элемент массива и поменять его местами с последним ( $N$ -м) элементом; выполнить описанные действия ( $N-1$ ) раз, каждый раз уменьшая на 1 количество анализируемых элементов и выводя содержимое массива.
2. Дан массив  $A$  размера  $N (< 6)$ . Упорядочить его по возрастанию методом сортировки *простыми вставками*: сравнить элементы  $A_1$  и  $A_2$  и, при необходимости меняя их местами, добиться того, чтобы они оказались упорядоченными по возрастанию; затем обратиться к элементу  $A_3$  и переместить его в левую (уже упорядоченную) часть массива, сохранив ее упорядоченность; повторить этот процесс для остальных элементов, выводя содержимое массива после обработки каждого элемента (от 2го до  $N$ -го). При выполнении описанных действий удобно использовать прием «барьера», записывая очередной элемент перед его обработкой в дополнительный элемент массива  $A_0$ .

Задание №4 по вариантам (двумерные массивы (матрицы)):

1. Даны целые положительные числа  $M$  и  $N$ . Сформировать целочисленную матрицу размера  $M \times N$ , у которой все элементы  $i$ -й строки имеют значение  $10i$  ( $i=1, \dots, M$ ).
2. Даны целые положительные числа  $M$  и  $N$ . Сформировать целочисленную матрицу размера  $M \times N$ , у которой все элементы  $j$ -го столбца имеют значение  $5j$  ( $j=1, \dots, N$ ).
3. Даны целые положительные числа  $M$ ,  $N$  и набор из  $M$  чисел. Сформировать матрицу размера  $M \times N$ , у которой в каждом столбце содержатся все числа из исходного набора (в том же порядке).
4. Даны целые положительные числа  $M$ ,  $N$  и набор из  $N$  чисел. Сформировать матрицу размера  $M \times N$ , у которой в каждой строке содержатся все числа из исходного набора (в том же порядке).
5. Дана матрица размера  $M \times N$  и целое число  $K$  ( $1 < K < M$ ). Вывести элементы  $K$ -й строки данной матрицы.
6. Дана матрица размера  $M \times N$  и целое число  $K$  ( $1 < K < N$ ). Вывести элементы  $K$ -го столбца данной матрицы.
7. Дана матрица размера  $M \times N$ . Вывести ее элементы, расположенные в строках с четными номерами (2, 4, ...). Вывод элементов производить по строкам, условный оператор не использовать.
8. Дана матрица размера  $M \times N$ . Вывести ее элементы, расположенные в столбцах с нечетными номерами (1, 3, ...). Вывод элементов производить по столбцам, условный оператор не использовать.
9. Дана матрица размера  $M \times N$ . Вывести ее элементы в следующем порядке: первая строка слева направо, вторая строка справа налево, третья строка слева направо, четвертая строка справа налево и т. д.
10. Дана матрица размера  $M \times N$ . Вывести ее элементы в следующем порядке: первый столбец сверху вниз, второй столбец снизу вверх, третий столбец сверху вниз, четвертый столбец снизу вверх и т. д.

Задание №5 по вариантам (двумерные массивы (матрицы)):

1. Дана матрица размера  $M \times N$  и целое число  $K$  ( $1 < K < M$ ). Найти сумму и произведение элементов  $K$ -й строки данной матрицы.
2. Дана матрица размера  $M \times N$  и целое число  $K$  ( $1 < K < N$ ). Найти сумму и произведение элементов  $K$ -го столбца данной матрицы.
3. Дана матрица размера  $M \times N$ . Для каждой строки матрицы найти сумму ее элементов.
4. Дана матрица размера  $M \times N$ . Для каждого столбца матрицы найти произведение его элементов.
5. Дана матрица размера  $M \times N$ . Для каждой строки матрицы с нечетным номером (1, 3, ...) найти среднее арифметическое ее элементов. Условный оператор не использовать.
6. Дана матрица размера  $M \times N$ . Для каждого столбца матрицы с четным номером (2, 4, ...) найти сумму его элементов. Условный оператор не использовать.
7. Дана матрица размера  $M \times N$ . В каждой строке матрицы найти минимальный элемент.
8. Дана матрица размера  $M \times N$ . В каждом столбце матрицы найти максимальный элемент.

9. Дана матрица размера  $M \times N$ . Найти номер ее строки с наибольшей суммой элементов и вывести данный номер, а также значение наибольшей суммы.
  10. Дана матрица размера  $M \times N$ . Найти номер ее столбца с наименьшим произведением элементов и вывести данный номер, а также значение наименьшего произведения.
  11. Дана матрица размера  $M \times N$ . Найти максимальный среди минимальных элементов ее строк.
  12. Дана матрица размера  $M \times N$ . Найти минимальный среди максимальных элементов ее столбцов.
  13. Дана матрица размера  $M \times N$ . В каждой ее строке найти количество элементов, меньших среднего арифметического всех элементов этой строки.
  14. Дана матрица размера  $M \times N$ . В каждом ее столбце найти количество элементов, больших среднего арифметического всех элементов этого столбца.
  15. Дана матрица размера  $M \times N$ . Найти номера строки и столбца для элемента матрицы, наиболее близкого к среднему значению всех ее элементов.
- Задание №6 по вариантам (двумерные массивы (матрицы)):

1. Дана квадратная матрица порядка  $M$ . Обнулить элементы матрицы, лежащие одновременно выше главной диагонали и выше побочной диагонали. Условный оператор не использовать.
2. Дана квадратная матрица порядка  $M$ . Обнулить элементы матрицы, лежащие одновременно выше главной диагонали и ниже побочной диагонали. Условный оператор не использовать.
3. Дана квадратная матрица порядка  $M$ . Обнулить элементы матрицы, лежащие одновременно ниже главной диагонали (включая эту диагональ) и выше побочной диагонали (также включая эту диагональ). Условный оператор не использовать.
4. Дана квадратная матрица порядка  $M$ . Обнулить элементы матрицы, лежащие одновременно ниже главной диагонали (включая эту диагональ) и ниже побочной диагонали (также включая эту диагональ). Условный оператор не использовать.

Порядок выполнения:

1. Запустить PascalABC.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 5. Символы и строки. Файлы**

Цель работы: получить навыки работы с символами, строками и файлами в PascalABC.

Интерактив – работа в малых группах.

Задания:

Задание №1 по вариантам (Символы и их коды. Формирование строк):

1. Дана непустая строка  $S$  и целое число  $N (> 0)$ . Вывести строку, содержащую символы строки  $S$ , между которыми вставлено по  $N$  символов «\*» (звездочка).
2. Дана непустая строка  $S$ . Вывести строку, содержащую символы строки  $S$ , между которыми вставлено по одному пробелу.
3. Дана строка. Вывести строку, содержащую те же символы, но расположенные в

обратном порядке.

4. Дано четное число  $N (>0)$  и символы  $C_1$  и  $C_2$ . Вывести строку длины  $N$ , которая состоит из чередующихся символов  $C_1$  и  $C_2$ , начиная с  $C_1$ .
5. Дано целое число  $N (>0)$  и символ  $C$ . Вывести строку длины  $N$ , которая состоит из символов  $C$ .
6. Дана непустая строка. Вывести коды ее первого и последнего символа.
7. Дано целое число  $N (1 < N < 26)$ . Вывести  $N$  последних *строчных* (то есть маленьких) букв латинского алфавита в обратном порядке (начиная с буквы «z»).
8. Дано целое число  $N (1 < N < 26)$ . Вывести  $N$  первых *прописных* (то есть заглавных) букв латинского алфавита.
9. Дан символ  $C$ . Вывести два символа, первый из которых предшествует символу  $C$  в кодовой таблице, а второй следует за символом  $C$ .
10. Дано целое число  $N (32 < N < 126)$ . Вывести символ с кодом, равным  $N$ .

Задание №2 по вариантам (Посимвольный анализ и преобразование строк):

1. Дана строка, изображающая целое положительное число. Вывести сумму цифр этого числа.
2. Дано целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке справа налево).
3. Дано целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке слева направо).
4. Дана строка. Если она представляет собой запись целого числа, то вывести 1, если вещественного (с дробной частью) — вывести 2; если строку нельзя преобразовать в число, то вывести 0. Считать, что дробная часть вещественного числа отделяется от его целой части десятичной *точкой* «.».
5. Дана строка. Преобразовать в ней все строчные буквы (как латинские, так и русские) в прописные, а прописные — в строчные.
6. Дана строка. Преобразовать в ней все строчные буквы (как латинские, так и русские) в прописные.
7. Дана строка. Преобразовать в ней все прописные латинские буквы в строчные.
8. Дана строка. Подсчитать общее количество содержащихся в ней строчных латинских и русских букв.
9. Дана строка. Подсчитать количество содержащихся в ней прописных латинских букв.
10. Дана строка. Подсчитать количество содержащихся в ней цифр.

Задание №3 по вариантам (Обработка строк с помощью стандартных функций):

В заданиях можно считать, что исходная строка не содержит *перекрывающихся* вхождений требуемых подстрок.

1. Даны строки  $S$  и  $S_0$ . Удалить из строки  $S$  все подстроки, совпадающие с  $S_0$ . Если совпадающих подстрок нет, то вывести строку  $S$  без изменений.
2. Даны строки  $S$  и  $S_0$ . Удалить из строки  $S$  последнюю подстроку, совпадающую с  $S_0$ . Если совпадающих подстрок нет, то вывести строку  $S$  без изменений.
3. Даны строки  $S$  и  $S_0$ . Удалить из строки  $S$  первую подстроку, совпадающую с  $S_0$ . Если совпадающих подстрок нет, то вывести строку  $S$  без изменений.
4. Даны строки  $S$  и  $S_0$ . Найти количество вхождений строки  $S_0$  в строку  $S$ .
5. Даны строки  $S$  и  $S_0$ . Проверить, содержится ли строка  $S_0$  в строке  $S$ . Если содержится, то вывести TRUE, если не содержится, то вывести FALSE.
6. Дан символ  $C$  и строки  $S$ ,  $S_0$ . После каждого вхождения символа  $C$  в строку  $S$  вставить строку  $S_0$ .
7. Дан символ  $C$  и строки  $S$ ,  $S_0$ . Перед каждым вхождением символа  $C$  в строку  $S$  вставить строку  $S_0$ .
8. Дан символ  $C$  и строка  $S$ . Удвоить каждое вхождение символа  $C$  в строку  $S$ .
9. Даны целые положительные числа  $N_1$  и  $N_2$  и строки  $S_1$  и  $S_2$ . Получить из этих строк новую строку, содержащую первые  $N_1$  символов строки  $S_1$  и последние  $N_2$  символов строки  $S_2$  (в указанном порядке).

10. Дано целое число  $N (>0)$  и строка  $S$ . Преобразовать строку  $S$  в строку длины  $N$  следующим образом: если длина строки  $S$  больше  $N$ , то отбросить первые символы, если длина строки  $S$  меньше  $N$ , то в ее начало добавить символы «.» (точка).

Задание №4 по вариантам (Анализ и преобразование слов в строке):

Во всех заданиях предполагается, что исходные строки являются непустыми и не содержат начальных и конечных пробелов.

1. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним пробелом и расположенные в алфавитном порядке.

2. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним пробелом и расположенные в обратном порядке.

3. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, заменив в нем все предыдущие вхождения его последней буквы на символ «.» (точка). Например, слово «МИНИМУМ» надо преобразовать в «.ИНИ.УМ». Количество пробелов между словами не изменять.

4. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, заменив в нем все последующие вхождения его первой буквы на символ «.» (точка). Например, слово «МИНИМУМ» надо преобразовать в «МИНИ.У». Количество пробелов между словами не изменять.

5. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Вывести строку, содержащую эти же слова, разделенные одним символом «.» (точка). В конце строки точку не ставить.

6. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого длинного слова.

7. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти длину самого короткого слова.

8. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат ровно три буквы «А».

9. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат хотя бы одну букву «А».

10. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые начинаются и заканчиваются одной и той же буквой.

Задание №5 по вариантам (Основные операции с текстовыми файлами):

1. Дано имя файла и целое число  $N (0 < N < 27)$ . Создать текстовый файл с указанным именем и записать в него  $N$  строк: первая строка должна содержать строчную (то есть маленькую) латинскую букву «а», вторая — буквы «ab», третья — буквы «abc» и т. д.; последняя строка должна содержать  $N$  начальных строчных латинских букв в алфавитном порядке.

2. Дан текстовый файл. Вывести количество содержащихся в нем символов и строк (маркеры концов строк EOLN и конца файла EOF при подсчете количества символов не учитывать).

3. Даны два текстовых файла. Добавить в конец первого файла содержимое второго файла.

4. Даны два текстовых файла. Добавить в начало первого файла содержимое второго файла.

5. Дано целое число  $K$  и текстовый файл. Вставить пустую строку после строки файла с номером  $K$ . Если строки с таким номером нет, то оставить файл без изменений.

6. Дан текстовый файл. Продублировать в нем все пустые строки.

7. Дано имя файла и целое число  $N (0 < N < 27)$ . Создать текстовый файл с указанным именем и записать в него  $N$  строк длины  $N$ ; строка с номером  $K (K = 1, \dots, N)$  должна содержать  $K$  начальных прописных (то есть заглавных) латинских букв, дополненных справа символами

«\*» (звездочка). Например, для  $N = 4$  файл должен содержать строки «A\*\*\*», «AB\*\*», «ABC\*», «ABCD».

8. Дана строка  $S$  и текстовый файл. Добавить строку  $S$  в конец файла.

9. Дана строка  $S$  и текстовый файл. Добавить строку  $S$  в начало файла.

10. Дано целое число  $K$  и текстовый файл. Вставить пустую строку перед строкой файла с номером  $K$ . Если строки с таким номером нет, то оставить файл без изменений.

Задание №6 по вариантам (Анализ и форматирование текста):

1. Дано целое число  $K$  и текстовый файл. Удалить из файла абзац с номером  $K$  (абзацы выделяются с помощью красной строки, 5 пробелов). Пустые строки между абзацами не учитывать и не удалять. Если абзац с данным номером отсутствует, то оставить файл без изменений.

2. Дан текстовый файл. Вывести первое слово текста наибольшей длины. Словом считать набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки.

3. Дано целое число  $K$  и текстовый файл. Создать строковый файл и записать в него все слова длины  $K$  из исходного файла. Словом считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки. Если исходный файл не содержит слов длины  $K$ , то оставить результирующий файл пустым.

4. Дан символ  $C$  — строчная (маленькая) русская буква и текстовый файл. Создать строковый файл и записать в него все слова из исходного файла, содержащие хотя бы одну букву  $C$  (прописную или строчную). Словом считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки. Если исходный файл не содержит подходящих слов, то оставить результирующий файл пустым.

5. Дан текстовый файл, содержащий текст, выровненный по левому краю. Выровнять текст по центру, добавив в начало каждой непустой строки нужное количество пробелов (ширину текста считать равной 50). Строки нечетной длины перед центрированием дополнять слева пробелом.

6. Дан текстовый файл. Найти количество абзацев в тексте, если первая строка каждого абзаца начинается с 5 пробелов («красная строка»). Пустые строки между абзацами не учитывать.

7. Дан текстовый файл. Абзацы выделяются в нем с помощью красной строки, 5 пробелов, а пустых строк нет. Вставить между соседними абзацами по одной пустой строке (в начало и конец файла пустые строки не добавлять).

8. Дан текстовый файл. Вывести последнее слово текста наименьшей длины. Словом считать набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки.

9. Дан символ  $C$  - прописная (заглавная) русская буква и текстовый файл. Создать строковый файл и записать в него все слова из исходного файла, начинающиеся на эту букву (прописную или строчную). Словом считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки. Если исходный файл не содержит подходящих слов, то оставить результирующий файл пустым.

10. Дан текстовый файл, содержащий текст, выровненный по левому краю. Выровнять текст по правому краю, добавив в начало каждой непустой строки нужное количество пробелов (ширину текста считать равной 50).

Порядок выполнения:

1. Запустить PascalABC.

2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.

2. Провести отладку и тестирование.

### Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

## **Лабораторная работа № 6. Пользовательские типы данных**

Цель работы: получить навыки работы с пользовательскими типами данных в PascalABC.

Интерактив – работа в малых группах.

### Задания:

Задание №1 по вариантам (Перечисляемый тип):

Определить среднюю зарплату за первое полугодие и среднюю зарплату за год.

Описание переменных:

months – перечисляемый тип;

month – очередной месяц;

hyear1, hyear2 – суммарный доход за 1 и 2 полугодия;

salary – зарплата в очередном месяце;

av1 – средняя зарплата за первое полугодие;

av2 – средняя зарплата за год.

Алгоритм решения задачи:

1. В цикле for переменной month по-порядку присваиваются все значения типа months.  
for month:=Jan to Dec do begin ...
2. Считывается значение зарплат в очередном месяце.
3. Если месяц принадлежит первому полугодию, то значение добавляется к первой сумме.
4. Если месяц принадлежит второму полугодию, то значение добавляется ко второй сумме.
5. По завершении цикла for вычисляются средние значения за полгода и год и выводятся на экран.

Задание №2 (Множества):

Проверить введенную пользователем строку на наличие недопустимых символов. В качестве первого символа допустимы только буквы и знак подчеркивания. Остальные символы могут быть буквами, цифрами и знаком подчеркивания.

Описание переменных:

str - строка, подлежащая проверке; вводится пользователем;

flag - переменная-флаг для хранения информации о том, корректна строка-ID или нет.

flag: boolean;

Тип и константы:

t\_chs - множество всех возможных символов;

lett - множество, содержащее английские буквы и знак подчеркивания;

num - множество, содержащее цифры.

Алгоритм решения задачи:

1. Если первый символ строки не принадлежит множеству lett, то flag устанавливается в false.

flag := false

str[1] in lett возвращает false, если символ не принадлежит множеству.

if not (str[1] in lett)...

Получается выражение not false, которое дает true, и ветка if выполняется. Если символ принадлежит lett, то выражение str[1] возвращает true, отрицание которого в свою очередь дает false.)

2. Иначе проверяем все символы строки, начиная со второго, на принадлежность объединению множеств lett и num.

if not (str[i] in lett+num)

Как только встречается символ, который не принадлежит данному объединению, то flag устанавливается в значение false и осуществляется прерывание цикла (break).

3. Если переменная flag имеет значение true, то выводится сообщение о корректном ID, иначе о некорректном.

Задание №3 по вариантам (Записи):

1. Опишите запись с именем типа Systema, содержащую информацию о планетах солнечной системы: номер планеты по удалению от Солнца (тип integer); название планеты (тип string); объем (real); диаметр (real); удаленность от Земли (real). Переменную, определяющую запись, назовите Planeta.

2. Опишите запись с именем типа Data, содержащую информацию о средней температуре в хранилище за 30 дней: номер месяца (тип integer); температура (тип real). Переменную, определяющую запись, назовите zamer. Без помощи with присвойте начальное значение: месяц - июль и температура для первого дня - 9,5.

3. Составьте программу, которая описывает таблицу химических элементов, отображая текущую информацию: название, символическое обозначение, массу атома, заряд атомного ядра, перечень основных химических свойств. Программа должна выполнять вывод данных о химическом элементе по указанному символическому обозначению, находить элемент с самой большой массой, с самым маленьким зарядом ядра.

4. Составьте программу, которая описывает массив записей жильцов дома, отображая в нем следующую информацию о каждом: номер квартиры, фамилия, имя, возраст, для лиц старше 18 лет в зависимости от рода занятий (учеба, работа, пенсия) - запись места учебы, места работы и трудового стажа, для пенсионеров - год выхода на пенсию. Программа должна обеспечивать ввод данных, поиск квартиры с максимальным числом жильцов, поиск самого юного и самого пожилого жильца, поиск студентов, пенсионеров.

5. Опишите, используя структуру записи, вступительные экзамены, на которых абитуриенты сдавали три экзамена, а для поступления надо было набрать 12 баллов. Составьте программу, считывающую с клавиатуры результаты всех вступительных экзаменов и выводящую на экран следующую информацию: список абитуриентов, сдавших все три экзамена на 5; список абитуриентов, потерпевших неудачу на экзаменах; список абитуриентов, зачисленных в институт.

6. Опишите, используя структуру записи, школьный журнал. Предусмотрите в записи поля для хранения информации о фамилии учащегося, предмете, оценке. Составьте программу, считывающую с клавиатуры данные об успеваемости учащихся класса и выводящую на экран сведения об отличниках класса, о средней успеваемости учащихся класса.

7. Опишите, используя структуру записи, класс или студенческую группу (фамилия, инициалы, дата рождения, месяц рождения, год рождения). Составьте программу, считывающую с клавиатуры данные об учащихся класса и выводящую на экран данные о днях рождения учащихся по месяцам.

8. В справочной службе аэропорта хранится расписание вылета самолетов на следующие сутки. Для каждого рейса указаны его номер, тип самолета, пункт назначения, время вылета. Определите все номера рейсов, типы самолетов и время их вылета для заданного пункта назначения.

9. У администратора железнодорожных касс хранится информация о свободных местах в поездах по всем направлениям на ближайшую неделю. Данная информация представлена в следующем виде: дата выезда, конечный пункт назначения, время отправления, число свободных купейных мест, число свободных плацкартных мест. Оргкомитет конференции обращается к администратору с просьбой зарезервировать 50 купейных мест до Москвы на субботу. При этом время отправления поезда должно быть не позднее 10 часов вечера. Выведите на экран время отправления или сообщение о невозможности выполнить заказ в полном объеме.

10. В радиоателье хранятся квитанции о сданной в ремонт радиоаппаратуре. Каждая квитанция содержит следующую информацию: наименование группы изделий (телевизор, радиоприемник и т. п.), марка изделия, дата приемки в ремонт, состояние готовности заказа (выполнен, не выполнен). Для директора ателье необходимо выдать информацию о состоя-



нии заказов на текущие сутки по группам изделий.

11. В библиотеке имеется список книг. Каждая запись этого списка содержит фамилии авторов, название книги, год издания. Определите, имеются ли в данном списке книги, в названии которых встречается некоторое ключевое слово (например, Pascal). Если такие книги представлены, то выведите на экран фамилии авторов, название и год издания этих книг. Предусмотрите ввод ключевого слова с клавиатуры.

12. В магазине имеется список поступивших в продажу автомобилей. Каждая запись этого списка содержит марку автомобиля и его параметры: стоимость, расход бензина на 100 км, надежность (число лет безотказной работы), комфортность (отличная, хорошая, удовлетворительная). Покупатель в свою очередь имеет ряд требований по каждому из этих параметров. Эти требования задаются в виде некоторого интервала (например, стоимость - 3...6 тыс. долл.; расход бензина - 410 л на 100 км). Выведите на печать перечень автомобилей, удовлетворяющих требованиям покупателя. Предусмотрите ввод требований покупателя с клавиатуры.

Задание №4 по четным или нет вариантам (Записи):

1. Описать тип TDate - запись с полями целого типа Day (день), Month (месяц) и Year (год) - и функцию LeapYear(D) с параметром типа TDate, которая возвращает 1, если год в дате D является високосным, и 0 в противном случае. Вывести значение функции LeapYear для пяти данных дат. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400.

2. Описать тип TPoint - запись с полями вещественного типа X и Y (координаты точки на плоскости) - и функцию Leng(A, B) вещественного типа, находящую длину отрезка AB на плоскости по координатам его концов:

$$|AB| = \sqrt{(A.X - B.X)^2 + (A.Y - B.Y)^2}$$

(A и B - параметры типа TPoint). С помощью этой функции найти длины отрезков AB, AC, AD, если даны координаты точек A, B, C, D.

Порядок выполнения:

1. Запустить PascalABC.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

## **Лабораторная работа № 7. Динамические структуры данных**

Цель работы: получить навыки работы с динамическими типами данных в PascalABC.

Интерактив – работа в малых группах.

Задания:

Задание №1 по вариантам (Односвязные списки):

Решите задачи Задания № 3 лабораторной работы № 6 «Пользовательские типы данных» с помощью односвязных списков.

Задание №2 по вариантам (Указатели):

1. Реализуйте с помощью указателей работу стека.
2. Реализуйте с помощью указателей работу очереди.

Стек реализует схему LIFO (*Last In, First Out – последний пришел, первый вышел*). Стек можно представить как стопку тарелок. Если положить тарелку на верх стопки, то мы будем иметь доступ только к этой тарелке. Реализовать стек достаточно просто – нужен массив и указатель на верхушку стека. Стек должен поддерживать 2 операции: push(x) – добавить x в стек, и pop() – извлечь верхушку из стека.

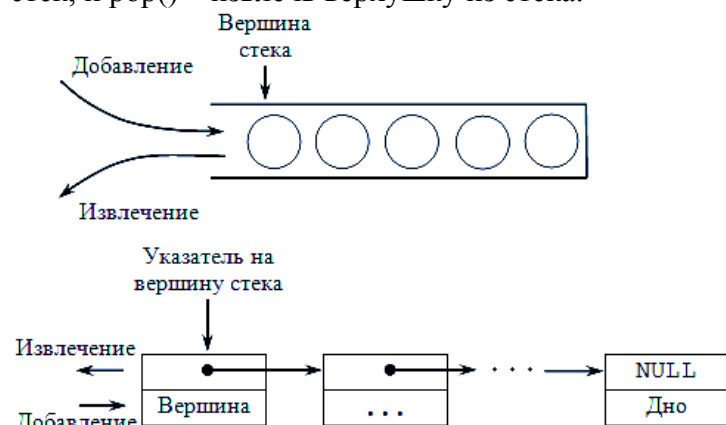


Рис. 7.14. Работа стека

Односторонняя очередь реализует схему FIFO (первый пришел, первый ушел). Для реализации очереди нужно помимо массива с данными 2 указателя (на голову и хвост). Хвост указывает на конец очереди, а голова на начало.

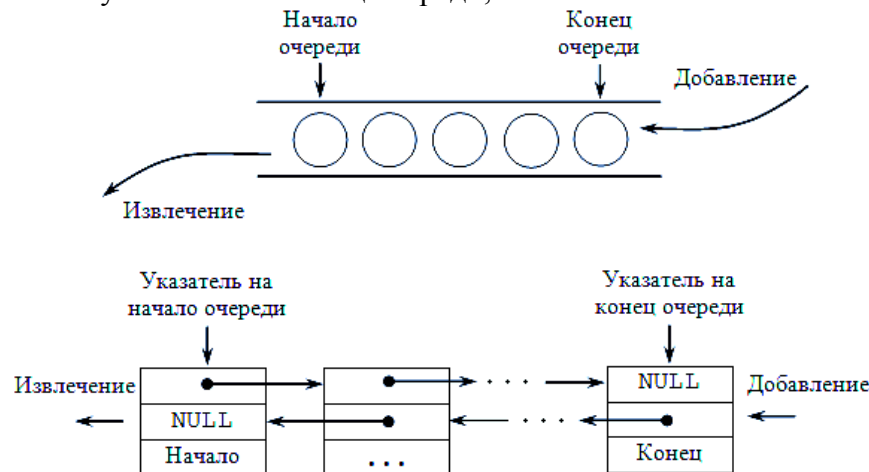


Рис. 7.15. Работа очереди

Задание №3 по вариантам (Динамические массивы):

1. Инициализировать массив строк размерностью n (n вводится пользователем). Сохранить массив в файл.
2. Инициализировать массив строк значениями из файла (см. задачу 1).

Порядок выполнения:

1. Запустить PascalABC.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.

2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 8. Объекты**

Цель работы: получить навыки работы с объектами в PascalABC.

Интерактив – работа в малых группах.

Задание:

Придумайте иерархию объектов из области, соответствующей вашему варианту, указав какие поля и методы есть у каждого класса.

- 1) шахматные фигуры;
- 2) геометрические фигуры;
- 3) вычислительная техника;
- 4) участники разработки программного средства: например, человек, заказчик, разработчик, контролер, пользователь;
- 5) состав персонала предприятия: например, руководитель, менеджеры, исполнители, рабочие;
- 6) мир животных;
- 7) мир птиц;
- 8) мир рыб;
- 9) мир простейших организмов;
- 10) документы: например, текстовые, аудио, видео.

Порядок выполнения:

1. Запустить PascalABC.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования PascalABC.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 9. Первый проект**

Цель работы: получить первичные навыки работы в Delphi Community Edition.

Задание:

1. Запустите Embarcadero RAD Studio XE7.
2. Выберите File → NewVCL Forms Application - Delphi.
3. Нажмите F12, чтобы перейти в редактор кода, и снова F12, чтобы вернуться обратно в конструктор формы.
4. Установите значения свойств стартовой формы в соответствии с таблицей 2.
5. Добавьте на форму два компонента Edit.
6. Добавьте на форму три компонента Label.
7. Добавьте на форму два компонента Button.
8. Создайте для кнопки Расчет (Button1) процедуру Button1Click обработки события Click.
9. Преобразуйте код процедуры.
10. Создайте для кнопки Завершить (Button2) процедуру Button2Click обработки события Click.
11. Преобразуйте код процедуры.

12. Прочтите пояснение к коду.
13. Сохраните проект в папке Pokupka.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 10. Внесение изменений и настройка**

Цель работы: научиться вносить изменения в проект.

Задание:

1. Внесите изменения в процедуру Button2Click в соответствии с приведенным ниже кодом. Система автоматически добавит слово finally вместо except.
2. Система автоматически добавит слово finally вместо except. Проверьте, как приложение будет реагировать на ошибки в обоих вариантах – со словом finally и со словом except.
3. Сохраните и закройте проект.
4. Откройте проект.
5. Создайте для поля Edit1 следующую процедуру Edit1KeyPress обработки события KeyPress.
6. Запустите приложение. Обратите внимание, что все-таки для корректной его работы требуется в последнюю очередь менять данные в поле Edit1, что не всегда удобно.
7. Создайте для поля Edit2 аналогичную процедуру Edit2Change обработки события Change.
8. Выберите Project → Options, чтобы открыть окно настроек Project Options.
9. В списке слева выберите вкладку Application.
10. Для установки режима компиляции в поле Target выберите Release configuration - 32-bit Windows platform.
11. Ниже в разделе Application Icon Settings в поле Icon: для изменения иконки приложения выберите Load Icon... при наличии .ico файла. Для возврата к системной иконке выберите Default.
12. Для задания названия приложения на вкладке Appearance (ниже вкладки Application) в поле Title введите, например, Покупка.
13. Нажмите ОК.
14. Для окончательной сборки приложения справа в окне Project Manager правой кнопкой мыши щелкните на пункте Release и в контекстном меню выберите команду Build.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

## Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 11. Form, Label и Button**

Цель работы: научиться создавать многооконное приложение.

#### Задание:

1. Создайте проект.
2. Поместите на форму компоненты и установите их свойства.
3. Чтобы добавить новую форму (Form2) к проекту выберете File→New→VCL Form – Delphi.
4. Установите свойство формы Caption в X-игрок, а свойство Position в poDesktopCenter и поместите на форму 9 кнопок Button.
5. Обратите внимание, что над формой имеются вкладки Unit1 и Unit2, по которым можно легко переходить между формами.
6. Добавьте к проекту еще одну форму (Form3), установите свойство формы Caption в O-игрок, а свойство Position в poDesktopCenter.
7. Перейдите по вкладке Unit2 к форме Form2, выделите и скопируйте все 9 кнопок, вернитесь обратно к форме Form3 и вставьте их на форму. Размеры формы Form3 сделайте как у формы Form2.
8. Сохраните проект.
9. Перейдите к форме Form1 и после слова implementation добавьте раздел uses и в нем пропишите ссылки на модули двух остальных форм:  
implementation  
uses  
Unit2, Unit3;
10. Перейдите к форме Form2 и также добавьте ссылки на модули Unit1 и Unit3, а для формы Form3 – ссылки на модули Unit1 и Unit2.
11. Перейдите к форме Form1 и дважды щелкните на кнопке Button1, чтобы создать для нее обработчик события Click.
12. Перейдите к форме Form2 и создайте для нее обработчик для события Close.
13. Перейдите к форме Form3 и создайте для нее обработчик для события Close.
14. Запустите приложение.
15. Перейдите к форме Form2 и создайте для нее константу C равную X.
16. Перейдите к форме Form3 и создайте для нее константу C равную O.
17. Перейдите к форме Form2 и создайте для Button1 обработчик события Click.
18. Аналогично создайте обработчики события Click и для остальных восьми кнопок. Код процедур будет тем же самым, что и для кнопки Button1, с тем лишь отличием, что название компонента будет соответствующим конкретной кнопки.
19. Создайте подобные обработчики и для формы Form3. Код процедур будет тем же самым, с тем лишь отличием, что вместо Form3 будет Form2.
20. Запустите приложение, чтобы убедиться в правильности вашего кода.
21. Перейдите к форме Form2 и в разделе private объявите функцию isWin.
22. После директивы {\$R \*.dfm} приведите код процедуры isWin.
23. Перейдите к форме Form3 и в разделе private объявите функцию isWin и после директивы {\$R \*.dfm} приведите код процедуры isWin.
24. Перейдите к форме Form2 и в разделе var объявите переменную btnN.
25. В разделе private объявите функцию winTest.
26. После кода процедуры isWin приведите код процедуры winTest.
27. Дополните коды процедур всех 9 кнопок вызовом функции winTest(btnN).

28. Перейдите к форме Form3 в разделе var объявите переменную btnN, в разделе private объявите функцию winTest, приведите ее код и дополните коды процедур всех 9 кнопок вызовом функции winTest(btnN).

29. Запустите приложение.

30. Сохраните проект.

Порядок выполнения:

1. Запустить Delphi Community Edition.

2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.

2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.

2. Назначение рассмотренных в работе операторов и функций.

3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 12. Проверочная работа №1**

Цель работы: закрепить навыки, полученные в лабораторных работах №№ 9, 10 и 11.

Задание:

Самостоятельно создать проект с изученными элементами управления.

Порядок выполнения:

1. Запустить Delphi Community Edition.

2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программы.

2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.

2. Назначение рассмотренных в работе операторов и функций.

3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 13. CheckBox, RadioButton и ComboBox**

Цель работы: получить первичные навыки работы с указанными компонентами.

Задания:

1. Создайте новый проект и добавьте на форму компоненты.

2. Для отображения фотографии добавьте компонент TImage, находящийся на вкладке Additional. Для свойства Picture щелкните на кнопку с тремя точками, в открывшемся окне нажмите на кнопку Load... и укажите путь к файлу с изображением, а затем нажмите на кнопку ОК.

3. Задайте компонентам CheckBox значения их свойств.

4. Создайте для кнопки ОК (Button1) процедуру Button1Click обработки события Click.

5. Создайте для компонента CheckBox1 процедуру CheckBox1Click обработки события Click.
6. Запустите приложение.
7. Создайте новый проект и добавьте на форму компоненты.
8. Для размещения нескольких компонентов TRadioButton добавьте компонент TGroupBox, находящийся там же на вкладке Standard.
9. После добавления компонента TGroupBox расположите на нем компоненты TRadioButton.
10. Задайте им значения их свойств.
11. Создайте для кнопки ОК (Button1) процедуру Button1Click обработки события Click.
12. Создайте для компонента Edit1 процедуру Edit1Change обработки события Change.
13. Создайте для компонента Edit1 процедуру Edit1KeyPress обработки события KeyPress.
14. Запустите приложение.
15. Создайте новый проект и добавьте на форму компоненты.
16. Создайте для кнопки ОК (Button1) процедуру Button1Click обработки события Click.
17. Создайте для формы (Form1) процедуру TForm1.FormCreate обработки события Create.
18. Запустите приложение.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 14. ListBox, BitButton и Image**

Цель работы: получить первичные навыки работы с указанными компонентами.

Задания:

1. Создайте новый проект и добавьте на форму компоненты и установите значение их свойств.
2. Объявите в разделе private класса TForm1 две процедуры – rightAnswer и wrongAnswer.
3. Объявите после директивы {\$R \*.dfm} три переменные – choice1, choice2, и tryNumber. Для этого создайте раздел var.
4. Создайте для формы Form1 процедуру FormCreate обработки события Create.
5. Создайте для компонента ListBox1 процедуру ListBox1Click обработки события Click.
6. Создайте для компонента ListBox2 процедуру ListBox2Click обработки события Click.
7. Создайте для кнопки Проверить (BitButton1) процедуру BitBtn1Click обработки события Click.
8. Создайте для кнопки BitButton2 процедуру BitBtn2Click обработки события Click.
9. Запустите проект.
10. Создайте новый проект и добавьте на форму компоненты.
11. Поместите в папку проекта хотя бы один файл .jpg.
12. Добавьте в директиву uses модуля формы ссылки на модули FileCtrl и Jpeg.
13. Создайте для формы (Form1) процедуру TForm1.FillListBox, поместив ее объявление в секцию Protected, а код после {\$R \*.dfm}.
14. Создайте для кнопки Папка (Button1) процедуру Button1Click обработки события Click.

15. Создайте для компонента ListBox1 процедуру ListBox1Click обработки события Click.
16. Создайте для формы (Form1) процедуру TForm1.FormCreate обработки события Create.
17. Запустите приложение.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 15. Timer, UpDown и ProgressBar**

Цель работы: получить первичные навыки работы с указанными компонентами.

Задания:

1. Создайте новый проект и добавьте на форму компоненты.
2. Создайте для кнопки Старт (Button1) процедуру Button1Click обработки события Click.
3. Создайте для кнопки Сброс (Button2) процедуру Button2Click обработки события Click.
4. Создайте для компонента Timer1 процедуру Timer1Timer обработки события Timer.
5. Поместите в область private объявления класса TForm1 объявления переменных s и m.
6. Запустите приложение.
7. Создайте новый проект и добавьте на форму компоненты.
8. Поместите в папку проекта файл ring.wav.
9. Добавьте в директиву uses модуля формы ссылки на модули System.DateUtils и Winapi.MMsystem.
10. Создайте в области var объявление переменной AlarmTime.
11. Создайте обработчики событий FormCreate, Timer1Timer, Button1Click, Edit1KeyPress и Edit2KeyPress.
12. Запустите приложение.
13. Создайте новый проект и добавьте на форму компоненты.
14. Добавьте в директиву uses модуля формы ссылки на модули Vcl.ComCtrls и Vcl.StdCtrls.
15. Поместите в область private объявления класса TForm1 объявления переменных sn, rem и p.
16. Создайте обработчики событий FormCreate, Timer1Timer и Edit1KeyPress.
17. Запустите приложение.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал



лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

#### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 16. Panel, SpeedButton и StatusBar**

Цель работы: получить первичные навыки работы с указанными компонентами.

#### Задания:

1. Создайте новый проект и добавьте на форму компонент Tpanel и установите значение его свойств.
2. Добавьте на Panel1 три кнопки TspeedButton и установите значение их свойств.
3. Добавьте на форму компонент TImage и установите значение его свойств.
4. Создайте обработчики событий FormCreate, SpeedButton1Click, SpeedButton2Click, SpeedButton3Click и FormResize.
5. Добавьте в директиву uses модуля формы ссылки на модули Vcl.Imaging.jpeg и Vcl.FileCtrl.
6. Поместите в область private объявления формы Form1 объявление следующих переменных.

```
...
private
  aPath: string; // каталог, который выбрал пользователь
  aSearchRec: TSearchRec; // информация о файле
  Pictures : TStringList; // список иллюстраций
  n: integer; // номер иллюстрации, отображаемой в данный момент
```

7. Измените код обработчиков событий.
8. Создайте новый проект и добавьте на форму компоненты
9. Создайте обработчики событий Form1Create, Form1Close, Button1Click, Edit1KeyPress, Edit1Change и Timer1Timer.

10. В разделе private объявите процедуру ItsOk.

```
private
  // процедура проверяет, угадал ли игрок число
  procedure ItsOk;
```

11. В раздел var внесите объявление следующих переменных.

```
comp: integer; // секретное число
rem: integer = 60; // остаток времени на выполнение задания
n: integer = 0; // сделано попыток
```

12. Запустите приложение.

#### Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

#### Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

#### Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

#### Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

#### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.

3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 17. Проверочная работа №2**

Цель работы: закрепить навыки, полученные в лабораторных работах №№ 13, 14, 15 и 16.

Задание:

Самостоятельно создать проект с изученными элементами управления.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программы.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 18. Memo, MainMenu, OpenFileDialog и SaveDialog**

Цель работы: получить первичные навыки работы с указанными компонентами.

Задания:

1. Создайте новый проект и добавьте на форму компонент TMemo.
2. Установите для TMemo1 свойство Align в alClient.
3. Создайте файл data.txt и поместите его в папку проекта Debug.
4. Добавьте после раздела var раздел const и внесите туда объявление константы TEXTFILE  
const  
TEXTFILE = 'data.txt';
5. Создайте для единственной формы Form1 процедуру TForm1.FormActivate обработки события Activate.
6. Создайте для единственной формы Form1 процедуру TForm1.FormCloseQuery обработки события CloseQuery.
7. Запустите приложение.
8. Создайте новый проект и добавьте на форму компонент TMemo.
9. Установите для TMemo1 свойство Align в alClient.
10. Добавьте на форму компонент MainMenu и настройте его.
11. Добавьте на форму компоненты OpenFileDialog и SaveDialog.
12. Установите для компонента OpenFileDialog свойство Filter в  
13. Текст (\*.txt)|\*.txt|Все (\*.\*)|\*.\*
14. В секции var объявите переменную FileName.  
var  
Form1: TForm1;  
FileName: string;
15. Создайте для формы процедуру FormCloseQuery обработки события CloseQuery.
16. Создайте обработчики события Click для компонентов N2, N3, N4 и N5.
17. Создайте самостоятельно обработчик события Click с выводом информации о вас для компонента N7.
18. Запустите приложение.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 19. TaskDialog, FileOpenDialog и FileSaveDialog**

Цель работы: получить первичные навыки работы с указанными компонентами.

Задания:

1. Создайте новый проект и добавьте на форму компоненты.
2. Создайте для кнопки ОК (Button1) процедуру Button1Click обработки события Click.
3. Создайте для компонента Edit1 процедуру EditChange обработки события Change.
4. Создайте для компонента Edit2 процедуру EditChange обработки события Change, т.е. пропишите то же имя, что и для компонента Edit1. Теперь одна процедура будет обрабатывать события Change для обоих компонентов.
5. Создайте для компонента Edit1 процедуру EditKeyPress обработки события KeyPress.
6. Создайте для компонента Edit2 процедуру EditKeyPress обработки события KeyPress, т.е. пропишите то же имя, что и для компонента Edit1.
7. Запустите приложение.
8. Создайте новый проект и добавьте на форму компоненты, начиная с Memo.
9. Создайте обработчики событий FormCloseQuery, FormActivate, N2Click, N3Click, N4Click и N5Click.
10. В секции private формы объявите переменную aFile и функцию ToFile:

```
private
  aFile: string; // файл, в котором надо
                 // сохранить документ
  function ToFile(var aFile: string):boolean;
```

11. Запустите приложение.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.

3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 20. Проверочная работа №3**

Цель работы: закрепить навыки, полученные в лабораторных работах №№ 18 и 19.

Задание:

Самостоятельно создать проект с изученными элементами управления.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программы.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 21. Графика. Часть 1**

Цель работы: получить первичные навыки работы с указанными компонентами.

Задания:

1. Создайте новый проект и добавьте на форму компонент PaintBox.
2. В секции private объявления класса формы Form1 пропишите  
private  
back: TBitmap; // фоновый рисунок
3. Из папки Icons скопируйте в папку проекта файл skylight.bmp.
4. Для работы с датой и временем подключите модуль DateUtils после директивы {\$R \* .dfm}.
5. Сразу за подключением создайте раздел var и объявите переменную-массив month:  
var  
month: array[1..12] of string =  
('января', 'февраля', 'марта',  
'апреля', 'мая', 'июня',  
'июля', 'августа', 'сентября',  
'октября', 'ноября', 'декабря');
6. Создайте обработчики событий FormCreate и PaintBox1Paint.
7. Запустите приложение.
8. Создайте новый проект и добавьте на форму компонент PaintBox.
9. После директивы {\$R \* .dfm} создайте раздел var и объявите переменную-массив st:  
var  
st: array[1..5] of string =  
('psSolid - сплошная',  
'psDash – штриховая',  
'psDot- пунктирная, короткие штрихи',  
'psDashDot - штрих-пунктирная',  
'psDashDotDot - штрих, два пунктира');
10. Создайте обработчик события PaintBox1Paint.
11. Запустите приложение.
12. Создайте новый проект и добавьте на форму компонент PaintBox.

13. Скопируйте в папку проекта файл kurs.txt.
14. После директивы {\$R \*.dfm} создайте раздел var и объявите переменную-массив kurs и переменную nrec:

```
var
  kurs: array[1..15] of real; // массив данных
  nrec: integer; // количество чисел,
    // прочитанных из файла
```

15. Создайте обработчики событий PaintBox1Paint и FormCreate.
16. Запустите приложение.
17. Создайте новый проект и добавьте на форму компонент PaintBox.
18. Создайте обработчик события PaintBox1Paint.
19. Запустите приложение.

#### Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

#### Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

#### Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

#### Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

#### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 22. Графика. Часть 2**

Цель работы: получить первичные навыки работы с указанными компонентами.

#### Задания:

1. Создайте новый проект и добавьте на форму компонент PaintBox.
2. Создайте обработчик события PaintBox1Paint.
3. Запустите приложение.
4. Создайте новый проект и добавьте на форму компонент PaintBox.
5. Скопируйте в папку проекта файл kurs.txt.
6. Объявите переменные.
7. Создайте обработчики событий PaintBox1Paint и FormCreate.
8. Запустите приложение.
9. Создайте новый проект и добавьте на форму компонент PaintBox.
10. Скопируйте в папку проекта файл data.txt.
11. Объявите переменные и константы.
12. Создайте обработчики событий PaintBox1Paint и FormCreate.
13. Запустите приложение.

#### Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

#### Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

#### Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

### Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

#### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **Лабораторная работа № 23. Графика. Часть 3**

Цель работы: получить первичные навыки работы с указанными компонентами.

#### Задания:

1. Создайте новый проект.
2. В секции private объявления класса формы Form1 пропишите private  
Sky: TBitmap; // небо  
Plane: TBitmap; // самолет
3. В папку проекта скопируйте plane.bmp и sky.bmp.
4. Создайте обработчики событий FormCreate и FormPaint.
5. Запустите приложение.
6. Создайте новый проект.
7. В секции private объявления класса формы Form1 пропишите private  
bm: TBitmap; // "плитка"
8. Скопируйте в папку проекта файл back.bmp.
9. Создайте обработчики событий FormCreate и FormPaint.
10. Запустите приложение.
11. Создайте новый проект и добавьте на форму компонент Timer.
12. После директивы {\$R \*.dfm} создайте раздел var и объявите переменные.
13. Создайте обработчики событий FormCreate и Timer1Timer.
14. Запустите приложение.
15. Создайте новый проект и добавьте на форму компонент Timer.
16. После директивы {\$R \*.dfm} создайте раздел var и объявите переменные.
17. Создайте обработчики событий Timer1Timer, FormKeyDown, FormKeyUp, FormPaint, FormKeyPress и FormCreate.
18. Запустите приложение.
19. Создайте новый проект и добавьте на форму компонент Timer.
20. В папку проекта скопируйте plane.bmp и sky.bmp.
21. В секции private объявления формы опишите переменные.
22. Создайте обработчики событий Timer1Timer, FormPaint и FormCreate.
23. Запустите приложение.

#### Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

#### Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

#### Задания для самостоятельной работы:

1. Проверить правильность работы программ.
2. Провести отладку и тестирование.

### Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

#### Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

#### **Лабораторная работа № 24. Проверочная работа №4**

Цель работы: закрепить навыки, полученные в лабораторных работах №№ 21, 22 и 23.

Задание:

Самостоятельно создать проект с изученными элементами управления.

Порядок выполнения:

1. Запустить Delphi Community Edition.
2. Выполнить задания.

Форма отчетности:

Отчет по лабораторной работе, скрепленный титульным листом. Отчет должен содержать название работы, цель, задание и результат выполнения задания.

Задания для самостоятельной работы:

1. Проверить правильность работы программы.
2. Провести отладку и тестирование.

Рекомендации по выполнению заданий и подготовке к лабораторной работе

При подготовке и выполнении лабораторной работы рекомендуется использовать материал лекций соответствующих разделов и литературу, предложенную для изучения данной дисциплины.

Контрольные вопросы для самопроверки

1. Особенности языка и среды программирования Delphi.
2. Назначение рассмотренных в работе операторов и функций.
3. Синтаксис рассмотренных в работе операторов и функций.

### **9.2. Методические указания по выполнению контрольной работы, РГР**

#### **Методические указания по выполнению контрольной работы**

##### **Порядок выполнения контрольной работы**

При выполнении контрольной работы обучающийся должен:

- совершенствовать теоретические знания по дисциплине «Технологии программирования»;
- продемонстрировать способность обобщать, систематизировать и анализировать информацию, необходимую для проведения исследования и решения поставленных задач;
- совершенствовать навыки работы с учебной, научной, справочной литературой.

Пояснительная записка к контрольной работе должна содержать следующие структурные элементы: титульный лист; содержание; введение; основные разделы работы; заключение; список использованных источников; приложение.

В элементе «Содержание» указываются все разделы пояснительной записки с указанием страниц.

**Во введении** к контрольной работе необходимо осветить следующие вопросы: актуальность темы исследования, современное состояние проблемы, цели и задачи выполнения контрольной работы, предмет и объект исследования.

**Основная часть** контрольной работы должна содержать краткое изложение особенностей решения поставленной задачи. В практическом разделе требуется выполнить практическое задание.

В **заключении** излагаются основные результаты проведенного исследования, оценивается успешность решения поставленных задач и степень достижения цели выполнения контрольной работы.

**Список использованных источников** должен включать в себя перечень литературных и других источников, действительно использованных при выполнении контрольной работы, и состоять не менее чем из 10 позиций.

В **приложении** помещается полный код разработанного приложения.

Выполнение контрольной работы должно начинаться с подбора и глубокого изучения литературных источников по теме работы. Ориентиром в этой части работы может служить список рекомендуемой литературы, приведенный ниже. Данный список содержит перечень основных литературных источников, имеющих в университетской библиотеке.

Важнейшим требованием, предъявляемым к контрольной работе, является самостоятельный характер ее выполнения. Контрольная работа выполняется аккуратно, без исправлений в виде пояснительной записки в строгом соответствии со стандартом ФГБОУ ВО «БрГУ».

### **Тематика контрольных работ**

#### **Варианты теоретических заданий для выполнения контрольной работы**

- Вариант 1. Эволюция технологий программирования
- Вариант 2. Структурное и объектно-ориентированное программирование
- Вариант 3. Проектирование ПО: декомпозиция системы
- Вариант 4. Средства поддержки проекта
- Вариант 5. Методы защиты программ и данных

#### **Практическое задание для выполнения контрольной работы**

Разработать законченное приложение на языке PascalABC, реализующее ввод данных от пользователя, преобразование данных, хранение данных и вывод результатов. Особое внимание уделить командному интерфейсу пользователя, названиям переменных и констант, а также комментариям в коде программы.

### **Методические указания по выполнению РГР**

#### **Порядок выполнения РГР**

При выполнении РГР обучающийся должен:

- совершенствовать теоретические знания по дисциплине «Технологии программирования»;
- продемонстрировать способность обобщать, систематизировать и анализировать информацию, необходимую для проведения исследования и решения поставленных задач;
- совершенствовать навыки работы с учебной, научной, справочной литературой.

Пояснительная записка к РГР должна содержать следующие структурные элементы: титульный лист; содержание; введение; основные разделы работы; заключение; приложение.

В элементе «Содержание» указываются все разделы пояснительной записки с указанием страниц.

**Во введении** к РГР необходимо осветить следующие вопросы: цели и задачи выполнения контрольной работы, предмет и объект исследования.

**Основная часть** РГР должна содержать краткое изложение особенностей решения поставленной задачи.

**В заключении** излагаются основные результаты проведенного исследования, оценивается успешность решения поставленных задач и степень достижения цели выполнения контрольной работы.

**Список использованных источников** должен включать в себя литературных и других источников, действительно использованных при выполнении контрольной работы.

**Приложения** помещаются в пояснительную записку к контрольной работе при необходимости. Это может быть справочная информация, различные нормативные документы, которые, по мнению автора контрольной работы, необходимы для иллюстрации или аргументации положений контрольной работы.

Выполнение РГР должно начинаться с подбора и глубокого изучения литературных источников по теме работы. Ориентиром в этой части работы может служить список рекомендуемой литературы, приведенный ниже. Данный список содержит перечень основных литературных источников, имеющих в университетской библиотеке.

Важнейшим требованием, предъявляемым к РГР, является самостоятельный характер ее выполнения. РГР выполняется аккуратно, без исправлений в виде пояснительной записки в строгом соответствии со стандартом ФГБОУ ВО «БрГУ».



## Тематика РГР

### Практическое задание для выполнения РГР

Разработать графический интерфейс пользователя в открытом средстве с учетом требований к графическому интерфейсу пользователю, рассмотренных в лекциях. Снабдить РГР структурой проекта приложения, изображениями форм проекта приложения, а также необходимыми пояснениями к иллюстративному материалу, объясняющие принятые решения.

### 10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

1. Microsoft Windows Professional 7.
2. Kaspersky Security.
3. LibreOffice.
4. PascalABC.
5. Delphi Community Edition.

### 11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

| <i>Вид занятия<br/>(Лк, ЛР, кр, РГР, СР)</i> | <i>Наименование аудитории</i> | <i>Перечень основного оборудования</i>   | <i>№ ЛР</i> |
|--|-------------------------------|--|-------------|
| Лк   | Мультимедийный класс          | Интерактивная доска SMART Board 680I со встроенным проектором UX60. ПК: AMD Athlon™7550 Dual-Core Processor 250 GHz/RAM 2Gb/HDD; Монитор Samsung 943N MY19LS | -           |
| ЛР   | Дисплейный класс              | Оборудование14-ПК: Процессор AMD A6-6400 APU; RAM 4 Gb; HDD 500 Gb; Монитор TFT 19 LG1953S-SF; Принтер: HP LaserJet P3005;                                   | №№ 1-24     |
| кр   | Дисплейный класс              | Оборудование14-ПК: Процессор AMD A6-6400 APU; RAM 4 Gb; HDD 500 Gb; Монитор TFT 19 LG1953S-SF; Принтер: HP LaserJet P3005;                                   | -           |
| РГР  | Дисплейный класс              | Оборудование14-ПК: Процессор AMD A6-6400 APU; RAM 4 Gb; HDD 500 Gb; Монитор TFT 19 LG1953S-SF; Принтер: HP LaserJet P3005;                                   | -           |
| СР   | Читальный зал №1              | Оборудование 10 ПК i5-2500/H67/4Gb (монитор TFT19 Samsung); принтер HP LaserJet P2055D   | -           |

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

### 1. Описание фонда оценочных средств (паспорт)

| № компетенции | Элемент компетенции  | Раздел                                  | Тема  | ФОС                                    |
|---------------|--|---|---|--|
| ПК-15         | способность участвовать в работах по доводке и освоению информационных технологий в ходе внедрения и эксплуатации информационных систем  | 1. Программное средство (ПС)            | 1.1. Надежное ПС как продукт технологии программирования  | Вопрос к зачету, экзаменационный билет |
| ПК-17         | способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент, банковские системы, безопасность информационных систем, управление технологическими процессами, механика, техническая физика, энергетика, ядерная энергетика, силовая электроника, металлургия, строительство, транспорт, железнодорожный транспорт, связь, телекоммуникации, управление инфокоммуникациями, почтовая связь, химическая промышленность, сельское хозяйство, текстильная и легкая промышленность, пищевая промышленность, медицинские и биотехнологии, горное дело, обеспечение безопасности подземных предприятий и производств, геология, нефтегазовая отрасль, геодезия и картография, геоинформационные системы, лесной комплекс, химико-лесной комплекс, экология, сфера сервиса, системы массовой информации, дизайн, медиаиндустрия, а также предприятия различного профиля и все виды деятельности в условиях экономики информационного общества | 2. Этапы разработки ПС                  | 1.2. Источники ошибок в ПС.<br>1.3. Общие принципы разработки ПС.<br>2.1. Внешнее описание ПС.<br>2.2. Функциональное описание ПС.<br>2.3. Архитектура ПС.<br>2.4. Структура программы и модульное программирование.<br>2.5. Программный модуль.<br>2.6. Тестирование и отладка ПС.<br>2.7. Обеспечение функциональности и надежности ПС.<br>2.8. Обеспечение качества ПС.<br>2.9. Документирование ПС.<br>2.10 Управление разработкой и аттестация ПС.   |  |
|               |  | 3. Компьютерная поддержка разработки ПС | 3.1. Объектный подход к разработке ПС.<br>3.3. Методы проектирования ПС   |  |
|               |  | 4. Разработка интерфейса ПС             | 3.2. CASE-средства<br>4.1. Проектирование человеко-машинного интерфейса.<br>4.2. Психология HCI.<br>4.3. Психология цветовосприятия<br>4.4. Визуальные атрибуты отображаемой информации. Композиция и организация экрана.<br>4.5. Разработка структуры диалога.<br>4.6. Окна как элементы графического интерфейса. Модели интерфейса.<br>4.7. Объектно-ориентированный подход к проектированию интерфейса.<br>4.8. Юзабилити-тестирование<br>4.9. Стандарты на интерфейсы.<br>4.10. Средства проектирования пользовательских интерфейсов. |  |
| ПК-32         | способность адаптировать приложения к изменяющимся условиям функционирования   |   |   |  |
| ОПК-1         | владение широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий   |   |   |  |

## 2. Вопросы к зачету

| №<br>п/п | Компетенции |  | ВОПРОСЫ К ЗАЧЕТУ   | Тема                         |
|----------|-------------|--|--|------------------------------|
|          | Код         | Определение  |  |                              |
| 1        | 2           | 3  | 4  | 5                            |
| 1.       | ПК-15       | способность участвовать в работах по доводке и освоению информационных технологий в ходе внедрения и эксплуатации информационных систем  | 1.1 Программное средство (ПС), надежность ПС, технология программирования.<br>1.2 Источники ошибок в ПС.<br>1.3 Жизненный цикл ПС.<br>1.4 Качество ПС.   | 1. Программное средство (ПС) |
| 2.       | ПК-17       | способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент, банковские системы, безопасность информационных систем, управление технологическими процессами, механика, техническая физика, энергетика, ядерная энергетика, силовая электроника, металлургия, строительство, транспорт, железнодорожный транспорт, связь, телекоммуникации, управление инфокоммуникациями, почтовая связь, химическая промышленность, сельское хозяйство, текстильная и легкая промышленность, пищевая промышленность, медицинские и биотехнологии, горное дело, обеспечение безопасности подземных предприятий и производств, геология, нефтегазовая отрасль, геодезия и картография, геоинформационные системы, лесной комплекс, химико-лесной комплекс, экология, сфера сервиса, системы массовой информации, дизайн, медиаиндустрия, а также предприятия различного профиля и все виды деятельности в условиях экономики информационного общества | 2.1 Внешнее описание ПС.<br>2.2 Архитектура ПС.<br>2.3 Методы разработки структуры программы.<br>2.4 Порядок разработки программного модуля.<br>2.5 Тестирование и отладка ПС: основные понятия, принципы и виды отладки ПС.<br>2.6 Обеспечение защищенности ПС.<br>2.7 Обеспечение легкости применения, эффективности и сопровождаемости ПС.<br>2.8 Обеспечение мобильности ПС.<br>2.9 Документы управления разработкой ПС.<br>2.10 Назначение и процессы управления разработкой ПС.<br>2.11 Аттестация ПС. | 2. Этапы разработки ПС       |
| 3.       | ПК-32       | способность адаптировать приложения к изменяющимся условиям функционирования   |  |                              |
| 4.       | ОПК-1       | владение широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий   |  |                              |

### 3. Экзаменационные вопросы

| №<br>п/п | Компетенции |  | ЭКЗАМЕНАЦИОННЫЕ<br>ВОПРОСЫ   | Тема  |
|----------|-------------|--|--|---|
|          | Код         | Определение  |  |   |
| 1        | 2           | 3  | 4  | 5   |
| 1.       | ПК-15       | способность участвовать в работах по доводке и освоению информационных технологий в ходе внедрения и эксплуатации информационных систем  | <p><b>1.1</b> Программное средство (ПС), надежность ПС, технология программирования.</p> <p><b>1.2</b> Источники ошибок в ПС.</p> <p><b>1.3</b> Жизненный цикл ПС.</p>   | 1. Программное средство (ПС)  |
| 2.       | ПК-17       | способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент, банковские системы, безопасность информационных систем, управление технологическими процессами, механика, техническая физика, энергетика, ядерная энергетика, силовая электроника, металлургия, строительство, транспорт, железнодорожный транспорт, связь, телекоммуникации, управление инфокоммуникациями, почтовая связь, химическая промышленность, сельское хозяйство, текстильная и легкая промышленность, пищевая промышленность, медицинские и биотехнологии, горное дело, обеспечение безопасности подземных предприятий и производств, геология, нефтегазовая отрасль, геодезия и картография, геоинформационные системы, лесной комплекс, химико-лесной комплекс, экология, сфера сервиса, системы массовой информации, дизайн, медиаиндустрия, а также предприятия различного профиля и все виды деятельности в условиях экономики информационного общества | <p><b>1.4</b> Качество ПС.</p> <p><b>2.1</b> Внешнее описание ПС.</p> <p><b>2.2</b> Архитектура ПС.</p> <p><b>2.3</b> Методы разработки структуры программы.</p> <p><b>2.4</b> Порядок разработки программного модуля.</p> <p><b>2.5</b> Тестирование и отладка ПС.</p> <p><b>2.6</b> Документы управления разработкой ПС.</p> <p><b>2.7</b> Назначение и процессы управления разработкой ПС.</p> <p><b>2.8</b> Аттестация ПС.</p> <p><b>3.1</b> Объектный подход к разработке ПС.</p> <p><b>3.2</b> Процессы ЖЦ ПС.</p> <p><b>3.3</b> Методологии IDEF.</p> <p><b>3.4</b> Инструменты разработки ПС.</p> <p><b>3.5</b> Инструментальные среды разработки и сопровождения ПС.</p> <p><b>3.6</b> Основные классы инструментальных сред разработки и сопровождения ПС.</p> <p><b>3.7</b> Основные изменения в жизненном цикле ПС для компьютерной технологии.</p> <p><b>3.8</b> Инструментальные системы технологии программирования.</p> <p><b>3.9</b> Подходы к проектированию и методологии разработки интерфейсов.</p> <p><b>4.1</b> Модели построения интерфейса</p> <p><b>4.2</b> Окна. Их виды и структура</p> <p><b>4.3</b> Рабочая область – контейнер. Рабочая книга. Проект</p> | <p>2. Этапы разработки ПС</p> <p>3. Компьютерная поддержка разработки ПС</p> <p>4. Разработка интерфейса ПС</p> |
| 3.       | ПК-32       | способность адаптировать приложения к изменяющимся условиям функционирования   |  |   |
| 4.       | ОПК-1       | владение широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий   |  |   |

#### 4. Описание показателей и критериев оценивания компетенций

| Показатели  | Оценка                     | Критерии   |
|---|----------------------------|--|
| <p><b>Знать:</b><br/> <b>ПК-15:</b><br/> – принципы обеспечения на стадии проектирования критериев качества программного средства облегчающих его доводку и освоение в ходе внедрения и эксплуатации;</p> <p><b>ПК-17:</b><br/> – методы разработки программного средства с помощью компьютерных технологий;</p> <p><b>ПК-32:</b><br/> – методы объектного подхода к разработке программного средства;</p> <p><b>ОПК-1:</b><br/> – цели и этапы разработки программного средства;</p> <p><b>Уметь:</b><br/> <b>ПК-15:</b><br/> – организовать разработку программного средства в соответствии с приоритетами обеспечения качества;</p> <p><b>ПК-17:</b><br/> – создавать и модифицировать проект в изучаемом CASE-средстве с использованием имеющихся в нем элементов управления;</p> <p><b>ПК-32:</b><br/> – применять изучаемые компьютерные технологии к адаптации программного средства к изменяющимся условиям функционирования;</p> <p><b>ОПК-1:</b><br/> – создавать и модифицировать программный код в соответствии с правилами изучаемого языка программирования;</p> <p><b>Владеть:</b><br/> <b>ПК-15:</b><br/> – навыками тестирования и отладки программного средства;</p> <p><b>ПК-17:</b><br/> – навыками объектно-ориентированного и визуального программирования;</p> <p><b>ПК-32:</b><br/> – навыками представления модели предметной области в терминах объектов;</p> <p><b>ОПК-1:</b><br/> – навыками структурного программирования.</p> | <b>зачтено</b>             | Оценка «зачтено» выставляется в случае, если студент демонстрирует: <ul style="list-style-type: none"> <li>– достаточно полное знание программного материала;</li> <li>– правильное выполнение типовых заданий, направленных на применение программного материала;</li> <li>– правильное применение основных положений программного материала.</li> </ul>                                    |
|   | <b>не зачтено</b>          | Оценка «не зачтено» выставляется в случае, если студент демонстрирует: <ul style="list-style-type: none"> <li>– существенные пробелы в знании программного материала;</li> <li>– принципиальные ошибки при выполнении типовых заданий, направленных на применение программного материала;</li> <li>– невозможность применения основных положений программного материала.</li> </ul>          |
|   | <b>отлично</b>             | Оценка «отлично» выставляется в случае, если студент демонстрирует: <ul style="list-style-type: none"> <li>– всестороннее систематическое знание программного материала;</li> <li>– правильное выполнение типовых заданий, направленных на применение программного материала;</li> <li>– правильное применение основных положений программного материала.</li> </ul>                         |
|   | <b>хорошо</b>              | Оценка «хорошо» выставляется в случае, если студент демонстрирует: <ul style="list-style-type: none"> <li>– недостаточно полное знание программного материала;</li> <li>– выполнение с несущественными ошибками типовых заданий, направленных на применение программного материала;</li> <li>– применение с несущественными ошибками основных положений программного материала.</li> </ul>   |
|   | <b>удовлетворительно</b>   | Оценка «удовлетворительно» выставляется в случае, если студент демонстрирует: <ul style="list-style-type: none"> <li>– частичное знание программного материала;</li> <li>– частичное выполнение типовых заданий, направленных на применение программного материала;</li> <li>– частичное применение основных положений программного материала.</li> </ul>                                    |
|   | <b>неудовлетворительно</b> | Оценка «неудовлетворительно» выставляется в случае, если студент демонстрирует: <ul style="list-style-type: none"> <li>– существенные пробелы в знании программного материала;</li> <li>– принципиальные ошибки при выполнении типовых заданий, направленных на применение программного материала;</li> <li>– невозможность применения основных положений программного материала.</li> </ul> |

## **5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и опыта деятельности**

Цель и задачи дисциплины «Технологии программирования» представлены в разделе 1 настоящей рабочей программы. Место дисциплины в структуре образовательной программы представлено в разделе 2 настоящей рабочей программы. Распределение объема дисциплины по формам обучения с указанием видов учебных занятий представлено в разделе 3 настоящей рабочей программы. Содержание дисциплины указано в разделе 4 настоящей рабочей программы.

Учебно-методические материалы для самостоятельной работы студентов по дисциплине находятся в свободном доступе в соответствии с разделом 6 настоящей рабочей программы.

При изучении дисциплины необходимо использовать литературу, указанную в разделе 7 настоящей рабочей программы, а также перечень ресурсов информационно-телекоммуникационной сети «Интернет», представленных в разделе 8 настоящей рабочей программы.

Консультации для студентов по дисциплине проводятся в соответствии с графиком проведения консультаций, представленном на стенде кафедры, за которой закреплена указанная дисциплина.

К экзамену допускаются студенты очной формы обучения, которые выполнили и оформили все лабораторные работы, а также выполнили, оформили и защитили контрольную работу и РГР. Методические указания по выполнению и оформлению представлены в разделе 9.1. настоящей рабочей программы. Методические указания по выполнению, оформлению и защите контрольной работы и РГР представлены в разделе 9.2. настоящей рабочей программы.

К экзамену допускаются студенты заочной формы обучения, которые выполнили и оформили все лабораторные работы, а также выполнили, оформили и защитили контрольную работу и РГР. Методические указания по выполнению и оформлению представлены в разделе 9.1. настоящей рабочей программы. Методические указания по выполнению, оформлению и защите контрольной работы и РГР представлены в разделе 9.2. настоящей рабочей программы.

Информационные технологии, используемые при освоении дисциплины, перечислены в разделе 10 настоящей рабочей программы.

Оценка знаний, умений, навыков осуществляется в процессе промежуточной аттестации обучающихся по дисциплине, которая осуществляется в виде зачета, экзамена. Для оценивания знаний, умений, навыков используются ФОС по дисциплине, содержащий экзаменационные билеты, вопросы к зачету.

Зачет проводится в письменной форме по выданному преподавателем заданию. Экзамен проводится в письменной форме по выданному преподавателем заданию.

По итогам выполненного задания преподаватель оценивает уровень знаний, умений, навыков. Описание показателей и критериев оценивания компетенций, сформированных по итогам изучения дисциплины, представлено в разделе 3 Приложения 1 настоящей рабочей программы. Основными оценочными средствами при проведении промежуточной аттестации являются экзаменационные билеты, вопросы к зачету и задания для лабораторных работ.

## АННОТАЦИЯ

### рабочей программы дисциплины Технологии программирования

#### 1. Цель и задачи дисциплины

Целью изучения дисциплины является: изучение основ современных технологий программирования и получение практических навыков их реализации.

Задачами изучения дисциплины является изучение структуры и общих свойств информации и информационных процессов, общих принципов построения вычислительных устройств, а также систем обработки, хранения и передачи информации.

#### 2. Структура дисциплины

2.1. Распределение трудоемкости по отдельным видам учебных занятий, включая самостоятельную работу: лекций – 53 часа, лабораторные работы – 70 часов, самостоятельная работа обучающихся – 129 часов.

Общая трудоемкость дисциплины составляет 288 часов, 8 зачетных единиц.

2.2. Основные разделы дисциплины:

- 1 – Программное средство (ПС).
- 2 – Этапы разработки ПС.
- 3 – Компьютерная поддержка разработки ПС.
- 4 – Разработка интерфейса ПС.

#### 3. Планируемые результаты обучения (перечень компетенций)

Процесс изучения дисциплины направлен на формирование следующих компетенций:  
ПК-15 - способность участвовать в работах по доводке и освоению информационных технологий в ходе внедрения и эксплуатации информационных систем;

ПК-17 - способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент, банковские системы, безопасность информационных систем, управление технологическими процессами, механика, техническая физика, энергетика, ядерная энергетика, силовая электроника, металлургия, строительство, транспорт, железнодорожный транспорт, связь, телекоммуникации, управление инфокоммуникациями, почтовая связь, химическая промышленность, сельское хозяйство, текстильная и легкая промышленность, пищевая промышленность, медицинские и биотехнологии, горное дело, обеспечение безопасности подземных предприятий и производств, геология, нефтегазовая отрасль, геодезия и картография, геоинформационные системы, лесной комплекс, химико-лесной комплекс, экология, сфера сервиса, системы массовой информации, дизайн, медиаиндустрия, а также предприятия различного профиля и все виды деятельности в условиях экономики информационного общества;

ПК-32 - способность адаптировать приложения к изменяющимся условиям функционирования;

ОПК-1 - владение широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий.

**4. Вид промежуточной аттестации:** зачет, экзамен.

*Протокол о дополнениях и изменениях в рабочей программе  
на 20\_\_-20\_\_ учебный год*

1. В рабочую программу по дисциплине вносятся следующие дополнения:

---

---

2. В рабочую программу по дисциплине вносятся следующие изменения:

---

---

---

Протокол заседания кафедры № \_\_\_\_ от «\_\_» \_\_\_\_\_ 20 \_\_\_\_ г.,  
*(разработчик)*

Заведующий кафедрой \_\_\_\_\_  
*(подпись)*

\_\_\_\_\_  
*(Ф.И.О.)*



Программа составлена в соответствии с федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.03 Информационные системы и технологии от «12» марта 2015 г. № 219

для набора 2015 года: и учебным планом ФГБОУ ВО «БрГУ» для очной формы обучения от «03» июля 2018 г. № 413.

для набора 2016 года: и учебным планом ФГБОУ ВО «БрГУ» для очной формы обучения от «16» сентября 2016 г. № 622, заочной формы обучения от «16» сентября 2016 г. № 622

**Программу составил:**

Горохов Д.Б., доцент каф. ИиПМ, к.т.н. \_\_\_\_\_

Рабочая программа рассмотрена и утверждена на заседании кафедры ИиПМ

от «19» декабря 2018 г., протокол № 5

И.о. заведующего кафедрой ИиПМ \_\_\_\_\_

А.С. Толстикова

**СОГЛАСОВАНО:**

Директор библиотеки \_\_\_\_\_

Т.Ф. Сотник

Рабочая программа одобрена методической комиссией ЕН факультета

от «20» декабря 2018 г., протокол № 4

Председатель методической комиссии факультета \_\_\_\_\_

М.А. Варданян

**СОГЛАСОВАНО:**

Начальник  
учебно-методического управления \_\_\_\_\_

Г.П. Нежевец

Регистрационный № \_\_\_\_\_