

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Узунов Федор Владимирович

Должность: Ректор

Дата подписания: 26.10.2021 14:05:36

Уникальный программный ключ:

fd935d10451b860e912264c0378f8448452bfdb603f94388008e29877a6bcbf5

**АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОРГАНИЗАЦИЯ  
«ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ ВЫСШЕГО ОБРАЗОВАНИЯ»  
«УНИВЕРСИТЕТ ЭКОНОМИКИ И УПРАВЛЕНИЯ»  
Факультет экономики и управления  
Кафедра «Бизнес-информатика»**

**УТВЕРЖДАЮ**

Проректор по учебно-методической работе

С.С. Скараник

«01» сентября 2020 г.



Рабочая программа дисциплины

**Объектно-ориентированный анализ и программирование**

Направление подготовки

**38.03.05 Бизнес-информатика**

Квалификация выпускника

*Бакалавр*

Для всех

форм обучения

Симферополь 2020

## Содержание

|  |    |
|--|----|
| 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы  | 3  |
| 2. Место дисциплины в структуре ОПОП бакалавриата  | 4  |
| 3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся      | 4  |
| 4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий  | 4  |
| 5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине  | 10 |
| 6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине  | 10 |
| 7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины  | 20 |
| 8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), необходимых для освоения дисциплины *   | 20 |
| 9. Методические указания для обучающихся по освоению дисциплины  | 21 |
| 10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости) | 21 |
| 11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине   | 21 |

# 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения ОПОП бакалавра обучающийся должен овладеть следующими результатами обучения по дисциплине

| Коды компетенций | Результаты освоения ОПОП  | Перечень планируемых результатов по дисциплине  |
|------------------|---|---|
| ОК-3             | способностью использовать основы экономических знаний в различных сферах деятельности   | <p><b><u>Знать:</u></b></p> <ul style="list-style-type: none"> <li>• основы экономических знаний в различных сферах деятельности;</li> <li>• Основные принципы самоорганизации и самообразования;</li> </ul>  |
| ОК-7             | способностью к самоорганизации и самообразованию  | <ul style="list-style-type: none"> <li>• инструментальные средства для обработки, анализа и систематизации информации по теме исследования;</li> <li>• методику подготовки научно-технических отчетов, презентации, научных публикаций по результатам выполненных исследований.</li> </ul>  |
| ПК-18            | способность использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования | <p><b><u>Уметь:</u></b></p> <ul style="list-style-type: none"> <li>• использовать основы экономических знаний в различных сферах деятельности</li> <li>• Организовать свое время, самостоятельно критически мыслить, формулировать свою точку зрения;</li> <li>• использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации;</li> <li>• готовить научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований.</li> </ul>   |
| ПК-19            | умение готовить научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований  | <p><b><u>Владеть:</u></b></p> <ul style="list-style-type: none"> <li>• способностью использовать основы экономических знаний в различных сферах деятельности;</li> <li>• Методами повышения квалификации, навыками накопления, обработки и использования информации;</li> <li>• способность использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования</li> <li>• умение готовить научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований.</li> </ul> |

## 2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина Б1.Б.18 «Объектно-ориентированный анализ» входит в базовую часть дисциплин. Ее изучению предшествуют такие дисциплины как: Информатика, математический анализ, базы данных. Дисциплина необходима для успешного изучения таких предметов как: проектирование ИС, web-программирование.

## 3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем и на самостоятельную работу обучающихся

Общая трудоёмкость дисциплины составляет 6 зачётных единицы 216 часов

### 1.1 Объем дисциплины по видам учебных занятий (в часах). Для очной формы обучения

Общая трудоёмкость дисциплины составляет 6 зачётных единицы 216 часов

| Объём дисциплины                           | Всего часов |
|--|-------------|
| Общая трудоемкость дисциплины              | 216         |
| Контактная работа                          | 118         |
| Аудиторная работа (всего):                 | 108         |
| Лекции                                     | 36          |
| Семинары, практические занятия             | 72          |
| Самостоятельная работа обучающихся (всего) | 98          |
| Зачет                                      | 4           |
| Экзамен                                    | 6           |

### Для заочной формы обучения

Общая трудоёмкость дисциплины составляет 6 зачётных единицы 216 часов

| Объём дисциплины                           | Всего часов |
|--|-------------|
| Общая трудоемкость дисциплины              | 216         |
| Контактная работа                          | 24          |
| Аудиторная работа (всего):                 | 14          |
| Лекции                                     | 8           |
| Семинары, практические занятия             | 6           |
| Самостоятельная работа обучающихся (всего) | 192         |
| Зачет                                      | 4           |
| Экзамен                                    | 6           |

## 4. Содержание дисциплины, структурированное по темам с указанием отведенного на них количества академических часов и видов учебных занятий

#### 4.1. Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)

| № темы   | Наименование темы   | Всего     |           | Количество часов                         |          |              |          |                  |           |
|--|---|-----------|-----------|--|----------|--------------|----------|------------------|-----------|
|  |   | ОФО       | ЗФО       | Контактная работа<br>(аудиторная работа) |          |              |          | Внеаудит. работа |           |
|  |   |           |           | Лекции                                   |          | Практические |          | Самост. работа   |           |
|  |   |           |           | ОФО                                      | ЗФО      | ОФО          | ЗФО      | ОФО              | ЗФО       |
| <b>Раздел 1. Теоретические основы объектно-ориентированного программирования</b> |   | <b>18</b> | <b>34</b> | <b>2</b>                                 | <b>2</b> | <b>2</b>     |          | <b>14</b>        | <b>32</b> |
| 1  | Теоретические основы объектно-ориентированного программирования | 18        | 34        | 2  | 2        | 2            |          | 14               | 32        |
| <b>Раздел 2. Основы программирования на ЯП</b>                                   |   | <b>34</b> | <b>36</b> | <b>6</b>                                 | <b>2</b> | <b>12</b>    | <b>2</b> | <b>16</b>        | <b>32</b> |
| 2  | Введение в визуальное программирование. Основы ЯП.              | 10        | 8         | 2  |          | 4            |          | 4                | 8         |
| 3  | Типы данных в ЯП  | 14        | 16        | 2  |          | 4            |          | 8                | 16        |
| 4  | Программирование линейных алгоритмов                            | 10        | 12        | 2  | 2        | 4            | 2        | 4                | 8         |
| <b>Раздел 3. Циклические алгоритмы и операторы выбора</b>                        |   | <b>36</b> | <b>36</b> | <b>6</b>                                 | <b>2</b> | <b>14</b>    | <b>2</b> | <b>16</b>        | <b>32</b> |
| 5  | Ветвящиеся алгоритмы  | 16        | 18        | 2  | 1        | 6            | 1        | 8                | 16        |
| 6  | Циклические алгоритмы   | 20        | 18        | 4  | 1        | 8            | 1        | 8                | 16        |
| <b>Раздел 4. Массивы, символы и строки.</b>                                      |   | <b>46</b> | <b>32</b> | <b>10</b>                                | <b>0</b> | <b>20</b>    | <b>0</b> | <b>16</b>        | <b>32</b> |
| 7  | Работа с символами  | 10        | 8         | 2  |          | 4            |          | 4                | 8         |
| 8  | Работа со строковым типом данных                                | 10        | 8         | 2  |          | 4            |          | 4                | 8         |
| <b>2 семестр</b>   |   |           |           |  |          |              |          |                  |           |
| 9  | Работа с массивами  | 16        | 8         | 4  |          | 8            |          | 4                | 8         |
| 10   | Комбинированные типы данных                                     | 10        | 8         | 2  |          | 4            |          | 4                | 8         |
| <b>Раздел 5. Работа с БД и файлами ЯП.</b>                                       |   | <b>52</b> | <b>40</b> | <b>10</b>                                | <b>0</b> | <b>22</b>    | <b>0</b> | <b>20</b>        | <b>40</b> |
| 11   | Теория проектирования баз данных                                | 10        | 8         | 2  |          | 4            |          | 4                | 8         |
| 12   | Приемы создания и модификации таблиц программно                 | 10        | 8         | 2  |          | 4            |          | 4                | 8         |
| 13   | Введение в клиент-серверные БД.                                 | 12        | 8         | 2  |          | 6            |          | 4                | 8         |
| 14   | Файлы. Файловые переменные.                                     | 10        | 8         | 2  |          | 4            |          | 4                | 8         |
| 15   | Технологические цепочки обработки файлов различных типов.       | 10        | 8         | 2  |          | 4            |          | 4                | 8         |

|                                     |                    |           |           |          |          |          |          |           |           |
|-------------------------------------|--------------------|-----------|-----------|----------|----------|----------|----------|-----------|-----------|
| <b>Раздел 6. История ЯП и ООАиП</b> |                    | <b>20</b> | <b>28</b> | <b>2</b> | <b>2</b> | <b>2</b> | <b>2</b> | <b>16</b> | <b>24</b> |
| 16                                  | История ЯП и ООАиП | 20        | 28        | 2        | 2        | 2        | 2        | 16        | 24        |
|                                     | Зачет, Экзамен     | 10        | 10        |          |          |          |          |           |           |
|                                     | <b>Итого</b>       | 216       | 216       | 36       | 8        | 72       | 6        | 98        | 192       |

## **4.2. Содержание дисциплины , структурированное по темам (разделам)**

### **Раздел 1. Теоретические основы объектно-ориентированного программирования**

Основные подходы к разработке программного обеспечения. Понятие объектно-ориентированного программирования (ООП). Понятие объекта и класса. Основные принципы ООП. Абстрагирование. Инкапсуляция. Агрегирование. Наследование. Объектная модель программы. Универсальный язык моделирования UML.

Цели и задачи дисциплины «Объектно-ориентированный анализ». Методология объектно-ориентированного программирования. Объектная модель.

### **Раздел 2. Основы программирования на языке ЯП**

Переменные объектных типов. Введение в визуальное программирование. Введение в ЯП. Первая программа и управление проектом. Создание программ для Windows. Начало работы. Форма. Стандартные компоненты. Страница Additional. Страница System.

Числовые типы данных. Символьный и строковый типы данных. Переменные и константы. Математические функции. Ввод данных. Вывод результатов. Вывод в поле диалогового окна.

### **Раздел 3. Циклические алгоритмы и операторы выбора**

Инструкция For. Инструкция While. Инструкция Repeat. Инструкция goto. Их свойства и отличия. Размещение компонентов CheckBox и RadioGroup на форме. Организация выбора одного из условий используя компонент CheckBox и RadioGroup. Инструкция if. Инструкция case.

### **Раздел 4. Массивы, символы и строки.**

Структурированные типы данных. Строковый тип данных. Строковые переменные. Строковые константы. Пустая строка. Выполнение операций над строками. Стандартные подпрограммы обработки строк.

Понятие массива данных. Линейный массив (вектор). Идентификация массива. Обращение к элементам массива. Заполнение массива: с клавиатуры, псевдослучайными данными; по закону последовательности. Константы-массивы. Типовые алгоритмы обработки массива. Вставка и удаление элементов массива. Двумерные массивы (матрицы).

Сортировка массивов. Поиск информации в упорядоченных и неупорядоченных структурах. Линейный и бинарный поиск.

Комбинированные типы данных: записи. Описание типа. Обращение к элементам типа «запись». Массивы записей.

### **Раздел 5. Работа с БД и файлами ЯП.**

Теория проектирования баз данных, ADO. Связь с таблицей MS Access, Поиск, фильтрация и индексация таблиц, Наборы данных. Основные свойства, методы и события, Таблицы Paradox в ADO, Поля (TField), Запросы, Приемы

создания и модификации таблиц программно, Сохранение древовидных структур в базе данных, Отчеты. Quick Report, Работа с сеткой DBGrid, DBChart. Графики и диаграммы, Введение в клиент-серверные БД. InterBase, Технические характеристики. Создание и перенос базы данных. Файлы. Файловые переменные. Стандартные (текстовые и бестиповые) и типизированные файлы. Технологические цепочки обработки файлов различных типов. Подпрограммы для работы с файлами различных типов.

#### **Раздел 6. История ЯП и ООАиП**

Обработка ошибок. Основные отличия различных версий Delphi. Другие языки объектно-ориентированного программирования

### **4.3. Темы практических занятий**

**Практическая работа № 1.** Изучение среды программирования

**Практическая работа № 2-3** Создания простой программы.

**Практическая работа № 4-5** Создание простых форм

**Практическая работа № 6-7.** Работа с линейными алгоритмами.

**Практическая работа № 8-9.** Программирование разветвляющихся алгоритмов.

**Практическая работа № 10.** Программирование алгоритмов с условием выбора.

**Практическая работа № 11-12.** Программирование циклических алгоритмов с предусловием.

**Практическая работа № 13-14.** Программирование циклических алгоритмов с постусловием.

**Практическая работа № 15-16.** Работа с символами

**Практическая работа № 17-18.** Работа со строками.

**Практическая работа № 19-20.** Работа с одномерными массивами

**Практическая работа № 21-22.** Работа с многомерными массивами

**Практическая работа № 23-24.** Работа с комбинированным типом данных

**Практическая работа №25-26.** Проектирование простой БД

**Практическая работа №27-28.** Создание таблиц в БД

**Практическая работа №29-30.** Клиент-серверные БД

**Практическая работа №31.** Графики и диаграммы в БД

**Практическая работа №32-33.** Работа с файлами.

**Практическая работа № 34-35.** Технологические цепочки обработки файлов разных типов

**Практическая работа №36.** Защита рефератов, индивидуальных проектов.

**5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

| Тема | Содержание заданий, выносимых на СРС                                    | Кол-во часов ОФО | Кол-во часов ЗФО | Учебно-методическое обеспечение |
|------|---|------------------|------------------|---------------------------------|
| 1    | Тема 1. Теоретические основы объектно-ориентированного программирования | 14               | 32               | Учебно-методическое пособие     |
| 2    | Тема 2. Основы программирования на ЯП                                   | 16               | 32               | Учебно-методическое пособие     |
| 3    | Тема 3. Циклические алгоритмы и операторы выбора                        | 16               | 32               | Учебно-методическое пособие     |
| 4    | Тема 4. Массивы, символы и строки.                                      | 16               | 32               | Учебно-методическое пособие     |
| 5    | Тема 5. Работа с БД и файлами ЯП.                                       | 20               | 40               | Учебно-методическое пособие     |
| 6    | Тема 6. История ЯП и ООАиП  | 16               | 24               | Учебно-методическое пособие     |

**6. Фонд оценочных средств, для проведения промежуточной аттестации обучающихся по дисциплине**

**6.1. Паспорт фонда оценочных средств по дисциплине**

**Компетенция ОК-3**

|  |   |   |
|--|---|---|
| способностью использовать основы экономических знаний в различных сферах деятельности  |   |   |
| <b>Этапы формирования компетенции</b>  |   |   |
| Знает  | Умеет   | Владеет   |
| основы экономических знаний в различных сферах деятельности<br>6.2.1(1-4,27)   | использовать основы экономических знаний в различных сферах деятельности<br>6.2.2,(2,4,9,13) 6.2.5(2) | способностью использовать основы экономических знаний в различных сферах деятельности<br>6.2.3,(5,2) 6.2.4(2) |
| <b>Показатели и критерии оценивания компетенции на различных этапах ее формирования, шкала оценивания</b>                        |   |   |
| Знает, если выполнил 6.2.1-1-4,27<br>Умеет, если выполнил 6.2.2,- 2,4,9,13 6.2.5-2<br>Владеет, если выполнил 6.2.3,- 5,2 6.2.4-2 |   |   |

**Компетенция ОК-7**

|  |   |   |
|--|---|---|
| способностью к самоорганизации и самообразованию     |   |   |
| <b>Этапы формирования компетенции</b>                |   |   |
| Знает  | Умеет   | Владеет   |
| Основные принципы самоорганизации и самообразования; | Организовать свое время, самостоятельно критически мыслить, | Методами повышения квалификации, навыками накопления, обработки и |

|   |  |   |
|---|--|---|
| 6.2.1(1-17)   | формулировать свою точку зрения 6.2.2,(1-3,5,7,12,20,22-32) 6.2.5(10-12) | использования информации 6.2.3,(1-5) 6.2.4(1-3) |
| <b>Показатели и критерии оценивания компетенции на различных этапах ее формирования, шкала оценивания</b> |  |   |
| Знает, если выполнил 6.2.1- 1-17  |  |   |
| Умеет, если выполнил 6.2.2,- 1-3,5,7,12,20,22-32 6.2.5-10-12  |  |   |
| Владеет, если выполнил 6.2.3,- 1-5 6.2.4-1-3  |  |   |

### Компетенция ПК-18

|   |  |   |
|---|--|---|
| способность использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования |  |   |
| <b>Этапы формирования компетенции</b>   |  |   |
| Знает   | Умеет  | Владеет   |
| инструментальные средства для обработки, анализа и систематизации информации по теме исследования;  | использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации | способность использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования |
| 6.2.1(18,22,23)   | 6.2.2,(6,8,10,11,14,17,18,19,21) 6.2.5(1,3-9)  | 6.2.3,(2,3,4) 6.2.4(1-3)  |
| <b>Показатели и критерии оценивания компетенции на различных этапах ее формирования, шкала оценивания</b>   |  |   |
| Знает, если выполнил 6.2.1-18,22,23   |  |   |
| Умеет, если выполнил 6.2.2,- 6,8,10,11,14,17,18,19,21 6.2.5-1,3-9   |  |   |
| Владеет, если выполнил 6.2.3,- 2,3,4 6.2.4-1-3  |  |   |

### Компетенция ПК-19

|  |   |   |
|--|---|---|
| умение готовить научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований |   |   |
| <b>Этапы формирования компетенции</b>  |   |   |
| Знает  | Умеет   | Владеет   |
| научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований                 | готовить научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований | способностью готовить научно-технические отчеты, презентации, научные публикации по результатам |
|  | 6.2.2,(6,8,10,11,14,17,18,19,21)  |   |

|   |              |  |
|---|--------------|--|
| 6.2.1(18,22,23)   | 6.2.5(1,3-9) | выполненных исследований<br>6.2.3,(2,3,4) 6.2.4(1-3) |
| <b>Показатели и критерии оценивания компетенции на различных этапах ее формирования, шкала оценивания</b> |              |  |
| Знает, если выполнил 6.2.1-18,22,23   |              |  |
| Умеет, если выполнил 6.2.2,- 6,8,10,11,14,17,18,19,21 6.2.5-1,3-9   |              |  |
| Владеет, если выполнил 6.2.3,- 2,3,4 6.2.4-1-3  |              |  |

## **6.2. Типовые контрольные задания или иные материалы**

### **6.2.1.1. Вопросы к зачету**

1. Основные понятия: объекты, компоненты, элементы управления.
2. Структура объекта. Требования к его описанию. Область действия полей.
3. Основы синтаксиса языка программирования. Простые и структурные типы данных.
4. Операторы языка программирования.
5. Структура программы. Характеристика разделов.
6. Среда разработки приложений. Меню, окна, панель инструментов.
7. Структура проекта. Характеристика файлов.
8. Класс окна форм: свойства, методы, события.
9. Взаимосвязь форм в приложении.
10. Классы командных кнопок (Button, BitBtn).
11. Классы флажков (CheckBox) .
12. Классы радиокнопок (RadioButton, RadioGroup, GroupBox).
13. Классы для работы с текстом (Label, Edit, Memo).
14. Классы списков (ListBox, ComboBox).
15. Стандартные визуальные компоненты.
16. Создание главного и всплывающего меню.
17. Компоненты страницы ADDITIONAL. Использование в приложениях.
18. Ввод в приложение контекстных подсказок.
19. Организация диалога с пользователем. Окно ввода.
20. Организация диалога с пользователем. Окна сообщений.
21. Компоненты Dialogs. Назначение и использование.
22. Календари, их использование в программных кодах.
23. Вставка графических объектов.
24. Разветвляющиеся алгоритмы. Инструкция if, case.
25. Циклические алгоритмы. Инструкция For, While.
26. Циклические алгоритмы. Инструкция Repeat, goto.
27. Математические функции в ЯП
28. Одномерные массивы в ЯП
29. Многомерные массивы в ЯП
30. Поиск элементов в массиве

### **6.2.1.2. типовые вопросы на экзамен (сообщения)**

1. Основные понятия: объекты, компоненты, элементы управления.
2. Структура объекта. Требования к его описанию. Область действия полей.
3. Основы синтаксиса языка программирования. Простые и структурные типы данных.
4. Операторы языка программирования.
5. Структура программы. Характеристика разделов.
6. Среда разработки приложений. Меню, окна, панель инструментов.
7. Структура проекта. Характеристика файлов.
8. Класс окна форм: свойства, методы, события.
9. Взаимосвязь форм в приложении.
10. Классы командных кнопок (Button, BitBtn).
11. Классы флажков (CheckBox).
12. Классы радиокнопок (RadioButton, RadioGroup, GroupBox).
13. Классы для работы с текстом (Label, Edit, Memo).
14. Классы списков (ListBox, ComboBox).
15. Стандартные визуальные компоненты.
16. Создание главного и всплывающего меню.
17. Компоненты страницы ADDITIONAL. Использование в приложениях.
18. Ввод в приложение контекстных подсказок.
19. Организация диалога с пользователем. Окно ввода.
20. Организация диалога с пользователем. Окна сообщений.
21. Компоненты Dialogs. Назначение и использование.
22. Календари, их использование в программных кодах.
23. Вставка графических объектов.
24. Разветвляющиеся алгоритмы. Инструкция if, case.
25. Циклические алгоритмы. Инструкция For, While.
26. Циклические алгоритмы. Инструкция Repeat, goto.
27. Математические функции в ЯП
28. Одномерные массивы в ЯП
29. Многомерные массивы в ЯП
30. Поиск элементов в массиве
31. Понятие качества программного средства.
32. Объектно-ориентированное программирование.
33. Понятие последовательного и прямого доступа в файлах.
34. Понятие файла. Текстовые файлы.
35. Стандартные процедуры и функции, применимые для любого типа файлов в языке высокого уровня.
36. Процедуры языка высокого уровня. Их описание и применение
37. Функции языка высокого уровня. Их описание и применение.
38. Строковый тип данных. Описание и операции над данными этого типа.
39. Подпрограммы языка высокого уровня. Их назначение.
40. Алгоритмы внутренней сортировки. Сортировка выбором.
41. Алгоритмы внутренней сортировки. Сортировка вставкой.
42. Алгоритмы внутренней сортировки. Сортировка обменом.
43. Логический тип данных.
44. Теория проектирования баз данных ADO при помощи ЯП.
45. Связь программируемой БД с таблицей MS Access.

46. Поиск, фильтрация и индексация таблиц при помощи ЯП
47. Уровни абстракции в процессе разработки программного обеспечения: архитектура, структура, реализация.
48. Цикл разработки прикладного программного обеспечения: концептуализация, анализ, проектирование, кодирование.
49. Цикл разработки прикладного программного обеспечения: тестирование, эволюция, сопровождение
50. Критерии оценки качества программы.
51. Тестирование программ.
52. Средства и инструменты разработки программного обеспечения.
53. Стиль программирования.
54. Отладка программного средства. Принципы и виды отладки.
55. Машинно-ориентированные языки. Языки высокого уровня.
56. Типовые этапы разработки алгоритмов.
57. Этапы решения задач на компьютере.
58. Представление о машинном языке. Пример программы на машинном языке для воображаемой машины.
59. Способы задания и описания алгоритмов.
60. Понятие исполнителя алгоритма

### **6.2.2. Типовые тесты**

#### **1. Легкость применения программного обеспечения это:**

- А) характеристики ПО, позволяющие минимизировать усилия пользователя по подготовке исходных данных, применению по;
- Б) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов;
- В) характеристики ПО, позволяющие минимизировать усилия по внесению изменений для устранения в нем ошибок и по его модификации.

#### **2. Мобильность программного обеспечения это:**

- А) способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей;
- Б) способность ПС безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени;
- В) способность ПО быть перенесенным из одной среды (аппаратного / программного) в другое.

#### **3. Устойчивость программного обеспечения — это:**

- А) свойство, характеризующее способность ПС завершать автоматически корректное функционирование ПК, несмотря на неправильные (ошибочные) входные данные;
- Б) свойство, способна противостоять преднамеренным или непреднамеренным деструктивным действиям пользователя;
- В) свойство, характеризующее способность ПС продолжать корректное функционирование, несмотря на неправильные (ошибочные) входные данные.

#### **4. UML — это:**

- А) язык программирования, имеющий синтаксис схожий с C ++;
- Б) унифицированный язык визуального моделирования, использует нотацию диаграмм;

В) набор стандартов и спецификаций качества программного обеспечения.

**5. При конструировании программного обеспечения на этапе разработки или выбора алгоритма решения реализуется следующее:**

- А) архитектурное обработки программы;
- Б) выбор языка программирования;
- В) совершенствование программы.

**6. Проектирование ПО в основном рассматривается как**

- А) архитектурное проектирование;
- Б) коммуникационные методы;
- В) детальные методы.

**7. На этапе тестирования пользователь выполняет следующее:**

- А) синтаксическое отладки;
- Б) выбор тестов и метода тестирования;
- В) определение формы выдачи результатов.

**8. Что из приведенного не является одним из методов проектирования программного обеспечения?**

- А) структурное программирование;
- Б) объектно-ориентированное программирование;
- В) алгебраическое программирования.

**9. Как называется процесс разбиения одной сложной задачи на несколько простых подзадач?**

- А) абстракция;
- Б) декомпозиция;
- В) реинжиниринг.

**10. Что из приведенного является критериями оценки удобства интерфейсов?**

- А) скорость обучения;
- Б) адаптация к стилю работы пользователя;
- В) все ответы правильные.

**11. Интерфейс пользователя — это**

- А) набор методов взаимодействия компьютерной программы и пользователя этой программы;
- Б) набор методов для взаимодействия между программами;
- В) способ взаимодействия между объектами.

**12. Интерфейс-это**

- А) прежде всего, набор правил;
- Б) набор задач пользователя, которые он решает с помощью системы;
- В) способ взаимодействия между объектами.

**13. Техническое задание — это**

- А) документ объяснений для заказчика;
- Б) исходный документ для сдачи ПО в эксплуатацию;
- В) выходной документ для проектирования, разработки автоматизированной системы.

**14. Архитектура программной системы —**

- А) декомпозиция решения для выделенного спектра задач домена на подсистемы или иерархию подсистем;

Б) определение системы в терминах вычислительных составляющих (подсистем) и интерфейсов между ними, которое отражает правила декомпозиции проблемы на составляющие;

В) соответствующие вариации состава выделенных компонент.

### **15. Агрегация —**

А) отношения, утверждает наличие связи между понятиями, не уточняя зависимости их содержания и объемов;

Б) возможность для некоторого класса находиться одновременно в связи с одним элементом из определенного множества классов;

В) объединение нескольких понятий в новое понятие, существенные признаки нового понятия при этом могут быть либо суммой компонент или существенно новыми (отношение «доля — целое»).

### **16. Ассоциация —**

А) возможность для некоторого класса находиться одновременно в связи с одним элементом из определенного множества классов;

Б) объединение нескольких понятий в новое понятие, существенные признаки нового понятия о этом могут быть либо суммой компонент или существенно новыми (отношение «доля — целое»);

В) самое общее отношение, утверждает наличие связи между понятиями, не уточняя зависимости их содержания и объемов.

### **17. Валидация -**

А) обеспечение соответствия разработки требованиям ее заказчиков.

Б) проверка правильности трансформации проекта в код реализации;

В) выявление всех ошибок.

### **18. Верификация —**

А) обеспечение соответствия разработки требованиям ее заказчиков;

Б) проверка правильности трансформации проекта в программу;

В) действия на каждой стадии жизненного цикла по проверке и подтверждению соответствия стандартам.

### **19. Процесс разработки ПО включает следующие процессы:**

А) сопровождения;

Б) проектирование;

В) эксплуатация.

### **20. Понятность — это**

А) атрибут функциональности, указывающий на возможность предотвращать несанкционированный доступ;

Б) атрибут надежности, который указывает на способность программы к перезапуску для повторного выполнения;

В) атрибут удобства, определяющий усилия, необходимые для распознавания логических концепций и условий их применения.

### **21. Артефакт — это**

А) любой продукт деятельности специалистов по разработке программного обеспечения;

Б) результат ошибок разработчика во входных или проектных спецификациях;

В) графическое представление элементов моделирования системы.

### **22. C#. Какой тип переменной используется в коде: `int a = 5;`**

- А) Знаковое 8-бит целое
- Б) Знаковое 64-бит целое
- В) Знаковое 32-бит целое

**23. C#. Что делает оператор «%»**

- А) Возвращает процент от суммы
- Б) Возвращает остаток от деления
- В) Возвращает тригонометрическую функцию
- Г) Ни чего из выше перечисленного.

**24. C#. Что сделает программа выполнив следующий код:**

**Console.WriteLine(«Hello, World!»);**

- А) Напишет на новой строчке Hello, World!
- Б) Напишет Hello, World!
- В) Удалит все значения с Hello, World!
- Г) Вырежет слово Hello, World! из всего текста

**25. C#. Как сделать инкрементацию числа**

- А) ++
- Б) —
- В) %%
- Г) !=

**26. C#. Как найти квадратный корень из числа x**

- А) Sqrt(x)
- Б) Summ.Koren(x);
- В) Arifmetic.sqrt(x);
- Г) Math.Sqrt(x);

**27. C#. Обозначения оператора «НЕ»**

- А) Not
- Б) No
- В) !
- Г) !=

**28. C#. бозначение оператора «ИЛИ»**

- А) !
- Б) !=
- В) ||
- Г) Or

**29. C#. Чему будет равен c, если int a = 10; int b = 4; bool c = (a == 10 && b == 4);**

- А) True
- Б) False
- В) Null
- Г) 14

**30. C#. Что вернет функция Termin после выполнения. Код:**

```
int Termin()
{
    int a = 1;
    int b = 3;
    if (a != 5) return a + b;
    else return 0;
```

- }  
 А) 5  
 Б) 3  
 В) 4  
 Г) 0

**31. С#. Как называется оператор «?:»**

- А) Вопросительный  
 Б) Прямой оператор  
 В) Тернарный оператор  
 Г) Территориальный оператор

**32. Что такое Куча ?**

- А) Это структура данных  
 Б) Именованная область памяти  
 В) Область динамической памяти  
 Г) Куча переменных

**КЛЮЧ К ТЕСТУ:**

|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| а | в | б | б | б | а | б | в | б | в  | а  | в  | в  | б  | в  | в  |

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| а  | б  | б  | в  | а  | в  | б  | а  | а  | Г  | в  | в  | а  | в  | в  | в  |

**6.2.3. Типовые практические задания**

**Задание 1.**

Составить на любом языке программирования консольное приложение, которое содержит описание класса **Time** (время), который должен содержать:

Класс должен включать:

- Закрытые свойства для хранения часов и минут
- Методы доступа к закрытым свойствам
- Конструктор или несколько конструкторов, для создания экземпляров класса
- Метод отображения на экране времени в формате (чч:мм)

Программа должна делать следующее:

1. В функции main() нужно объявить и создать массив из 3 объектов описанного класса
2. Задать им следующие значения (2ч 30м, 5ч 15м, 3ч 45м)
3. Вывести на экран время, хранящееся во всех объектах.
4. Рассчитать разницу в днях между 1 и 2 объектами и вывести ее на экран.

**Задание 2.**

Составить на любом языке программирования консольную программу, которая содержит описание класса

Date - дата (год, месяц, день)

Класс должен включать:

- Закрытые свойства для хранения год, месяц, день.
- Методы доступа к закрытым свойствам.
- Конструктор или несколько конструкторов, для создания объектов

класса.

- Метод - показать на экране время в формате (дд/мм/гг)
- Метод - рассчитать количество дней с начала года до даты

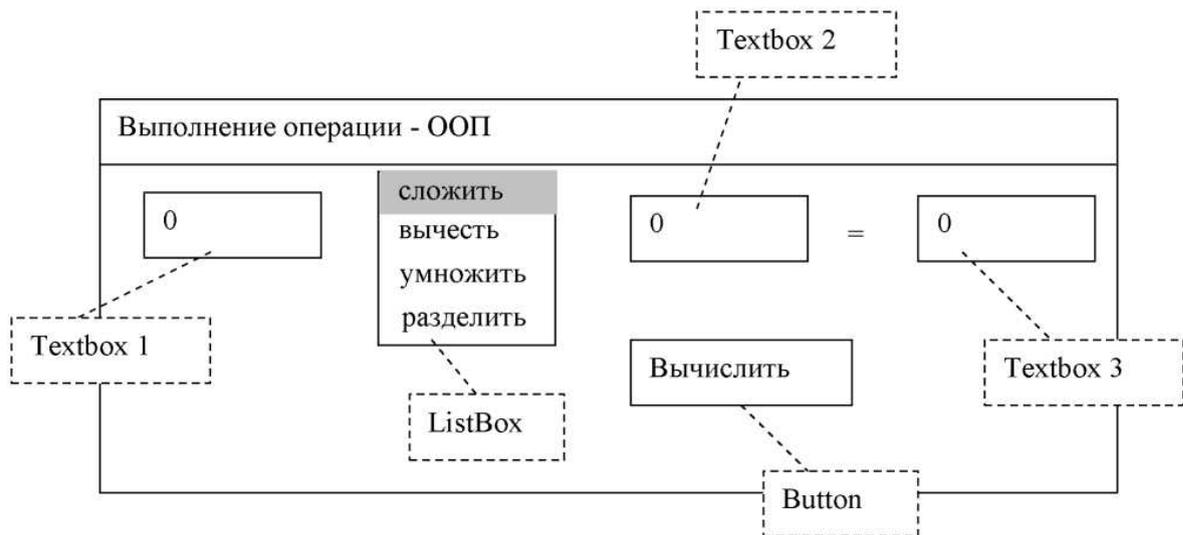
```
public int Days()
```

Программа должна делать следующее:

1. В функции main() нужно объявить и создать массив из 3 объектов описанного класса
2. Задать им следующие значения (1.5.2001 5.2.2002 13.7.2001)
3. Вывести на экран даты, хранящиеся во всех объектах.
4. Рассчитать разницу в днях между 1 и 3 объектами и вывести ее на экран.

**Задание 3.**

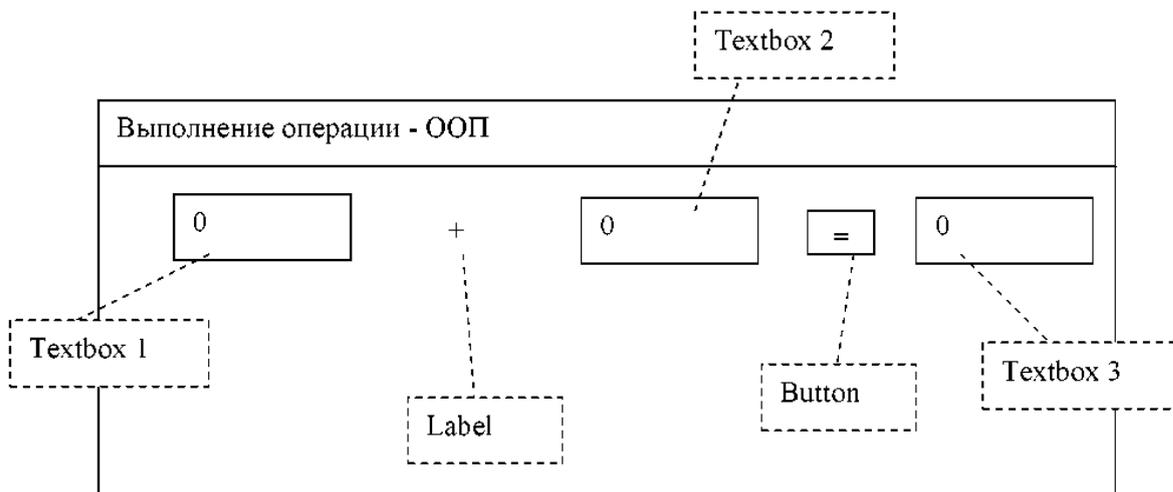
1. Написать программу, которая, используя объект класса производного от класса Form, выводит следующее окно:



При нажатии кнопки Button, текстовое поле Textbox3 должно показать результат выбранной в ListBox операции над числами, введенными в Textbox1 и Textbox2

**Задание 4.**

1. Написать программу, которая, используя объект класса производного от класса Form, выводит следующее окно:



При нажатии кнопки Button, текстовое поле Textbox3 должно показать результат выбранной в ListBox операции над числами, введенными в Textbox1 и Textbox2.

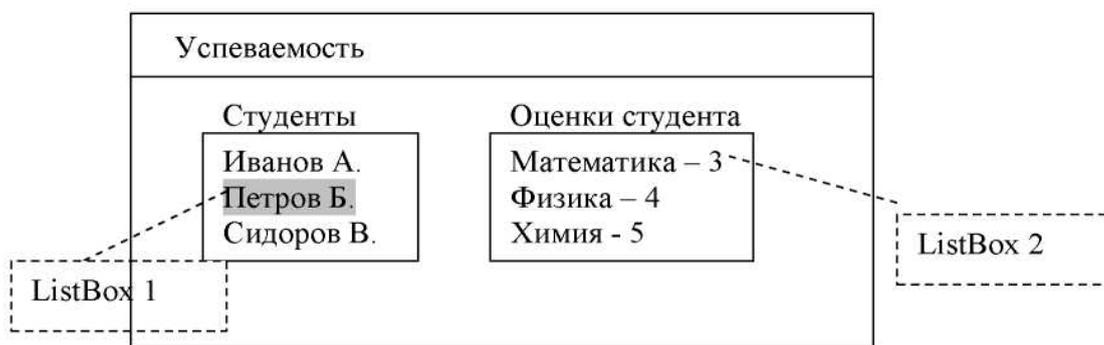
При выборе студента в ListBox1 в ListBox2 должны показываться его оценки.

### Задание 5.

1. Написать программу, которая, используя объект класса производного от класса Form, позволяет просматривать информацию об успеваемости студентов по годам.

| ФИО студента | Математика | Физика | Химия |
|--------------|------------|--------|-------|
| Иванов А.    | 4          | 5      | 4     |
| Петров Б.    | 3          | 4      | 5     |
| Сидоров В.   | 5          | 5      | 5     |

Форма окна программы:



При выборе студента в ListBox1 в ListBox2 должны показываться его оценки.

## 6.2.4. Типовые контрольные задания

### Контрольное задание 1

1. Поколения языков программирования. Их особенности.
2. Объектно-ориентированное проектирование.
3. Дать определение конструктору, деструктору. В чем отличие

конструктора от других методов в языке C#. Написать код, моделирующий некоторый класс Project, у которого есть атрибуты: name, duration, cost. Показать пример создания экземпляра класса Project с учетом вышеперечисленных параметров.

### **Контрольное задание 2**

1. Абстрагирование.
2. Объектно-ориентированный анализ.
3. Дать определение модификаторам доступа в языке C#. Создать класс Student с полями name, age, place\_born. Добавить метод, выводящий все информацию о конкретном студенте. Показать на примере процесс создания экземпляра класса Student и вывода информации о нем (с использованием ранее созданного метода).

### **Контрольное задание 3**

1. Объектно-ориентированное программирование.
2. Инкапсуляция.
3. Дать определение методу-свойству. Особенность стратегий реализации. Привести код для класса Mobile, имеющий атрибуты: model и color. Создать 2 метода-свойства для работы вышеперечисленными полями, причем один из них умеет только записывать значения, другой как записывать, так и считывать.

## **6.2.5. Темы рефератов**

1. Объектно-ориентированный анализ.
2. Предпосылки появления и развития объектно-ориентированного подхода в программировании.
3. Основные особенности объектно-ориентированного программирования.
4. Ключевые понятия объектно-ориентированного программирования.
5. Наследование и полиморфизм позднего связывания.
6. Объектная модель.
7. Внутренняя структура объекта.
8. Конструкторы и деструкторы, их назначение и правила использования.
9. Виртуальные и динамические методы, их назначение и механизмы вызова.
10. Особенности платформы .NET.
11. Функциональные типы в ЯП высокого уровня.
12. Обобщения в ЯП высокого уровня.

## **7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

### **а) основная литература:**

1. Бабушкина, И. А. Практикум по объектно-ориентированному программированию / И. А. Бабушкина, С. М. Окулов. — 5-е изд. — Москва : Лаборатория знаний, 2020. — 367 с. — ISBN 978-5-00101-780-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/12254.html> (дата обращения: 22.06.2020). — Режим доступа: для авторизир. пользователей

2. Лебедева, Т. Н. Теория и практика объектно-ориентированного

программирования : учебное пособие / Т. Н. Лебедева. — 2-е изд. — Челябинск, Саратов : Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа, 2019. — 221 с. — ISBN 978-5-4486-0663-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/81498.html> (дата обращения: 22.06.2020). — Режим доступа: для авторизир. пользователей

3. Мейер, Б. Объектно-ориентированное программирование и программная инженерия / Б. Мейер. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. — 285 с. — ISBN 978-5-4486-0513-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/79706.html> (дата обращения: 22.06.2020). — Режим доступа: для авторизир. пользователей

**б) дополнительная литература:**

4. Новиков, П. В. Объектно-ориентированное программирование : учебно-методическое пособие к лабораторным работам / П. В. Новиков. — Саратов : Вузовское образование, 2017. — 124 с. — ISBN 978-5-4487-0011-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/64650.html> (дата обращения: 22.06.2020). — Режим доступа: для авторизир. пользователей

5. Букунов, С. В. Основы объектно-ориентированного программирования : учебное пособие / С. В. Букунов, О. В. Букунова. — Санкт-Петербург : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2017. — 196 с. — ISBN 978-5-9227-0713-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/74339.html> (дата обращения: 22.06.2020). — Режим доступа: для авторизир. пользователей

**8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

1. <http://www.intuit.ru/> - материалы Национального открытого университета «ИНТУИТ»

2. Материалы по языку программирования C#:  
<http://msdn.microsoft.com/vcsharp/>  
<http://www.c-sharpcorner.com/>  
<http://www.csharp-station.com/>

**9. Методические указания для обучающихся по освоению дисциплины**

При проведении лекций, практических занятий, самостоятельной работе студентов применяются интерактивные формы проведения занятий с целью погружения студентов в реальную атмосферу профессионального сотрудничества по разрешению проблем, оптимальной выработки навыков и качеств будущего специалиста. Интерактивные формы проведения занятий предполагают обучение в сотрудничестве. Все участники образовательного процесса (преподаватель и студенты) взаимодействуют друг с другом,

обмениваются информацией, совместно решают проблемы, моделируют ситуацию.

В учебном процессе используются интерактивные формы занятий:

1. Творческое задание. Выполнение творческих заданий требуют от студента воспроизведение полученной ранее информации в форме, определяемой преподавателем, и требующей творческого подхода.

2. Групповое обсуждение. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и способствует лучшему усвоению изучаемого материала.

#### **10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)**

В процессе лекционных и практических занятий используется следующее программное обеспечение:

- программы, обеспечивающие доступ в сеть Интернет (например, «Google Chrome»);

- программы, демонстрации видео материалов (например, проигрыватель «Windows Media Player»);

- программы для демонстрации и создания презентаций (например, «Microsoft PowerPoint»);

- интегрированные среды программирования (например, Visual Studio Community);

- SQL-сервер и интегрированная среда управления инфраструктурой SQL (например, SQL Server Express и SQL Server Management Studio).

#### **11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине**

Для преподавания дисциплины требуются специальные материально-технические средства (компьютерный класс). Во время лекционных занятий, которые проводятся в большой аудитории, используется проектор для демонстрации слайдов, схем, таблиц и прочего материала.