



Программу составил(и):

*кфмн, Доцент, Бужан Виталий Викторович*

Рецензент(ы):

*дтн, профессор кафедры информационных систем и программирования КубГТУ, Видовский Л.А.; директор АО «ЮГ-СИСТЕМА ПЛЮС», Глебов О.В.*

Рабочая программа дисциплины

**Технологии программирования**

разработана в соответствии с ФГОС ВО:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 10.03.01 Информационная безопасность (приказ Минобрнауки России от 17.11.2020 г. № 1427)

составлена на основании учебного плана:

10.03.01 Информационная безопасность

утвержденного учёным советом вуза от 20.11.2023 протокол № 3.

Рабочая программа одобрена на заседании кафедры

**Кафедра математики и вычислительной техники**

Протокол от 13.10.2023 г. № 3

Зав. кафедрой Исикова Наталья Павловна

Согласовано с представителями работодателей на заседании НМС, протокол № 3 от 20.11.2023.

Председатель НМС проф. Павелко Н.Н.

<b>1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)</b>	
1.1	освоение студентами теоретических и практических основ проектирования программных систем
Задачи: приобретение знаний в области проектирования программных систем, современных компьютерных и программных средств проектирования программного обеспечения (ПО), изучения критериев качества программных продуктов.	
<b>2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ</b>	
Цикл (раздел) ОП:	Б1.О
<b>2.1</b>	<b>Требования к предварительной подготовке обучающегося:</b>
2.1.1	Структуры и алгоритмы обработки данных
2.1.2	Основы программирования
2.1.3	Безопасность систем баз данных
2.1.4	Интеллектуальные системы и технологии
2.1.5	Сети и телекоммуникации
2.1.6	Основы национальной безопасности
2.1.7	Теория систем и системный анализ
2.1.8	Основы информационной безопасности
2.1.9	Методы и средства криптографической защиты информации
<b>2.2</b>	<b>Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:</b>
2.2.1	Производственная практика: Преддипломная практика
2.2.2	Выполнение и защита выпускной квалификационной работы
<b>3. ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ, ИНДИКАТОРЫ ИХ ДОСТИЖЕНИЯ и планируемые результаты обучения</b>	
<b>УК-1: Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач</b>	
<b>УК-1.1: Анализирует задачу, выделяя ее базовые составляющие</b>	
<b>Знать</b>	
Уровень 1	Минимальный необходимый уровень знаний для анализа задач, с выделением ее базовых составляющих
Уровень 2	Уровень знаний для анализа задач, с выделением ее базовых составляющих в объеме, соответствующем программе подготовки, допущено несколько негрубых ошибок
Уровень 3	Уровень знаний для анализа задач, с выделением ее базовых составляющих в объеме, соответствующем программе подготовки, без ошибок
<b>УК-1.2: Определяет и ранжирует информацию, требуемую для решения поставленной задачи</b>	
<b>Уметь</b>	
Уровень 1	Продемонстрированы основные умения определять и ранжировать информацию, требуемую для решения поставленной задачи, решены типовые задачи с негрубыми ошибками, выполнены все задания, но не в полном объеме
Уровень 2	Продемонстрированы все основные умения определять и ранжировать информацию, требуемую для решения поставленной задачи, решены типовые задачи с негрубыми ошибками, выполнены все задания, но некоторые с недочётами
Уровень 3	Продемонстрированы все основные умения определять и ранжировать информацию, требуемую для решения поставленной задачи, решены типовые задачи с отдельными несущественными недочётами, выполнены все задания в полном объеме
<b>УК-1.3: Осуществляет поиск информации для решения поставленной задачи по различным типам запросов</b>	
<b>Владеть</b>	
Уровень 1	Имеется минимальный набор навыков осуществлять поиск информации для решения поставленной задачи по различным типам запросов с негрубыми ошибками и некоторыми недочётами
Уровень 2	Продемонстрированы базовые навыки осуществлять поиск информации для решения поставленной задачи по различным типам запросов с некоторыми недочётами
Уровень 3	Продемонстрированы навыки использования осуществлять поиск информации для решения поставленной задачи по различным типам запросов без ошибок и недочётов
<b>ОПК-7: Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности;</b>	
<b>ОПК-7.1: Разрабатывает с помощью языков высокого уровня алгоритмы решения типовых профессиональных задач</b>	
<b>Знать</b>	
Уровень 1	Минимальный необходимый уровень знаний разработки с помощью языков высокого уровня алгоритмы

	решения типовых профессиональных задач
Уровень 2	Уровень знаний разработки с помощью языков высокого уровня алгоритмы решения типовых профессиональных задач в объёме, соответствующем программе подготовки, допущено несколько негрубых ошибок
Уровень 3	Уровень знаний разработки с помощью языков высокого уровня алгоритмы решения типовых профессиональных задач в объёме, соответствующем программе подготовки, соответствующем программе подготовки, без ошибок
<b>ОПК-7.2: Разрабатывает программы для работы с файлами как с источником данных</b>	
<b>Уметь</b>	
Уровень 1	Продемонстрированы основные умения разрабатывать программы для работы с файлами как с источником данных, решены типовые задачи с негрубыми ошибками, выполнены все задания, но не в полном объёме
Уровень 2	Продемонстрированы все основные умения разрабатывать программы для работы с файлами как с источником данных, решены все основные задачи с негрубыми ошибками, выполнены все задания в полном объёме, но некоторые с недочётами
Уровень 3	Продемонстрированы все основные умения разрабатывать программы для работы с файлами как с источником данных, решены все основные задачи с отдельными несущественными недочётами, выполнены все задания в полном объёме
<b>ОПК-7.3: Отлаживает разработанные программные средства</b>	
<b>Владеть</b>	
Уровень 1	Имеется минимальный набор навыков по отладке разработанных программных средств с негрубыми ошибками и некоторыми недочётами
Уровень 2	Продемонстрированы базовые навыки по отладке разработанных программных средств с некоторыми недочётами
Уровень 3	Продемонстрированы навыки по отладке разработанных программных средств без ошибок и недочётов

#### 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература и эл. ресурсы	Практ . подг.
<b>Раздел 1. Программное обеспечение</b>						
1.1	Понятие программного обеспечения /Лек/	6	4	УК-1.1 УК-1.2 УК-1.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
1.2	Понятие программного обеспечения /Ср/	6	4	УК-1.1 УК-1.2 УК-1.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
1.3	Классификация программного обеспечения /Лек/	6	4	УК-1.1 УК-1.2 УК-1.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
1.4	Классификация программного обеспечения /Ср/	6	4	УК-1.1 УК-1.2 УК-1.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
<b>Раздел 2. Структурное программирование</b>						
2.1	Анализ программ /Лек/	6	2	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
2.2	Анализ программ /Ср/	6	2	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
2.3	Корректность и устойчивость программ /Лек/	6	2	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	

2.4	Корректность и устойчивость программ /Ср/	6	4	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
<b>Раздел 3. Объектно-ориентированное программирование</b>						
3.1	Варианты использования /Лек/	6	2	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
3.2	Варианты использования /Пр/	6	6	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
3.3	Варианты использования /Ср/	6	10	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
3.4	Объектное проектирование /Лек/	6	2	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
3.5	Объектное проектирование /Пр/	6	10	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
3.6	Объектное проектирование /Ср/	6	9,8	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
<b>Раздел 4. Компонентно-ориентированное программирование</b>						
4.1	Компоненты и компонентно-ориентированная модель конструирования программных продуктов /Лек/	7	10	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
4.2	Компоненты и компонентно-ориентированная модель конструирования программных продуктов /Пр/	7	8	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
4.3	Компоненты и компонентно-ориентированная модель конструирования программных продуктов /Ср/	7	12	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
4.4	Мультиплатформенное программное обеспечение /Лек/	7	8	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	
4.5	Мультиплатформенное программное обеспечение /Пр/	7	8	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6	

4.6	Мультиплатформенное программное обеспечение /Ср/	6	6	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
4.7	Платформа .NET /Лек/	7	8	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
4.8	Платформа .NET /Пр/	7	8	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
4.9	Платформа .NET /Ср/	7	16	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
4.10	Исполняемая и динамическая сборки .NET /Лек/	7	6	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
4.11	Исполняемая и динамическая сборки .NET /Пр/	7	8	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
4.12	Исполняемая и динамическая сборки .NET /Ср/	7	16	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
<b>Раздел 5. Промежуточная аттестация</b>					
5.1	Зачёт /КА/	6	0,2	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
5.2	Консультация /Консл/	7	1	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6
5.3	Экзамен /КАЭ/	7	0,3	УК-1.1 УК-1.2 УК-1.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8 Л1.9Л2.1 Л2.2 Л2.3 Э1 Э2 Э3 Э4 Э5 Э6

## 5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

### 5.1. Контрольные вопросы и задания

Примерный перечень вопросов к зачёту

1. Понятие технологии программирования, её цели и задачи.
2. Понятие программного обеспечения
3. Классификация программного обеспечения.
4. «Стихийное» программирование.
5. Структурный подход к программированию.
6. Понятие восходящего проектирования.
7. Понятие нисходящего проектирования.
8. Понятие программного модуля.

9. Модульное программирование.
10. Объектно-ориентированный подход к программированию.
11. Понятие объекта в объектно-ориентированном программировании.
12. Понятие класса объектов.
13. Наследование классов.
14. Перегрузка методов.
15. Понятие конструктора в объектно-ориентированном программировании.
16. Понятие свойства в объектно-ориентированном программировании.
17. Средства обработки данных объекта.
18. Компонентно-ориентированный подход к программированию.
19. Понятие программного компонента.
20. Основные понятия СОМ технологий.
21. Интерфейс СОМ-объектов.
22. Идентификаторы, используемые в СОМ технологии.
23. Управляющие элементы ActiveX.
24. Case-технологии.
25. Понятие интегрированной среды разработки программного обеспечения.

#### Примерный перечень вопросов к экзамену

1. Понятие технологии программирования, её цели и задачи.
2. Понятие программного обеспечения и его классификация.
3. Понятие системного программного обеспечения и его состав.
4. Понятие прикладного программного обеспечения и его состав.
5. Понятие инструментария программного обеспечения и его состав.
6. Понятие объекта в объектно-ориентированном подходе к программированию.
7. Понятие класса в объектно-ориентированном подходе к программированию.
8. Понятие инкапсуляции в объектно-ориентированном программировании.
9. Понятие наследования в объектно-ориентированном программировании.
10. Понятие полиморфизма в объектно-ориентированном программировании.
11. Понятие агрегации.
12. Перегрузка операций как разновидность полиморфизма в объектно-ориентированном подходе к программированию.
13. Перегрузка функций как разновидность полиморфизма в объектно-ориентированном подходе к программированию.
14. Сущность и назначение виртуальных функций.
15. Понятие абстрактных классов.
16. Определение интерфейса в объектно-ориентированном программировании.
17. Понятие ассоциации в объектно-ориентированном программировании.
18. Понятие зависимости в объектно-ориентированном программировании.
19. Понятие жизненного цикла программы.
20. Понятие структурного подхода к программированию.
21. Понятие технологии разработки программного обеспечения снизу-вверх.
22. Понятие технологии разработки программного обеспечения сверху-вниз.
23. Постановка задачи и спецификация программы. Способы записи алгоритма.
24. Цели и назначение технологии СОМ.
25. Понятие программного компонента.
26. Платформа .NET.
27. Виртуальная машина Java.
28. Понятие сборки. Исполняемые и динамические сборки.
29. Понятие распределённой сборки.
30. Назначение JIT-компилятора.
31. Назначение библиотек FCL.
26. Понятие, назначение и способ получения IL-кода

#### Список заданий по дисциплине

Задача 1. Разработать программу «Функциональный калькулятор». Про-грамма должна производить вычисление элементарных функций. Предусмотреть возможность добавления новых функций.

Задача 2. Создать класс CFile, инкапсулирующий в себе такие функции работы с файлами, как открытие, чтение, сохранение. На базе этого класса создать производный класс CMyDataFile — файл, содержащий в себе данные некоторого определённого типа MyData, а также заголовок, облегчающий до-ступ к этому файлу. Написать программу, демонстрирующую работу с этими классами. Программа должна содержать меню, позволяющее осуществить проверку всех методов классов.

Задача 3. Создать класс CPoint — точка. На его основе создать классы CcoloredPoint и CLine. На основе класса CLine создать класс CPolyLine — многоугольник. Класс должен иметь методы для установки и получения зна-чений всех координат. Написать демонстрационную программу, в которой будет использоваться список объектов этих классов в динамической памяти.

Задача 4. Разработать программу, осуществляющую перевод из одной системы координат в другую. Например, из декартовой в полярную систему координат, из сферической в декартову и наоборот. Предусмотреть добавление новых систем координат.

Задача 5. Разработать программу, осуществляющую статистический анализ массивов числовых данных с помощью специальных методов (вычислений средних величин, математического ожидания, корреляционного и регрессионного анализа и т. д.). Предусмотреть возможность добавление новых методов в разрабатываемый проект.

Задача 6. Разработать программу, реализующую операции линейной алгебры (сложение и произведение матриц). Предусмотреть возможность добавление новых операций в разрабатываемый проект.

Задача 7. Разработать программу, реализующую методы решений алгебраических уравнений (метод половинного деления). Предусмотреть возможность добавление новых методов в разрабатываемый проект.

Задача 8. Разработать программу, осуществляющую перевод из десятичной системы счисления в двоичную. Предусмотреть добавление новых систем счисления.

Задача 9. Разработать программу, осуществляющую вычисление объёмов пространственных фигур (куб, шар). Предусмотреть добавление новых типов фигур.

#### Тесты

1. Какие программы можно отнести к системному ПО  
А. драйверы  
В. текстовые редакторы  
С. электронные таблицы  
D. графические редакторы  
E. все ответы верны
2. Специфические особенности ПО как продукта  
А. продажа по ценам ниже себестоимости (лицензирование)  
В. низкие материальные затраты при создании программ  
С. возможность создание программ небольшим коллективом или даже одним человеком  
D. разнообразие решаемых задач с помощью программных средств  
E. все ответы верны
3. Какие программы нельзя отнести к системному ПО  
А. игровые программы  
В. компиляторы языков программирования  
С. операционные системы  
D. системы управления базами данных  
E. все ответы верны
4. Специфические особенности ПО как продукта  
А. низкие затраты при дублировании  
В. универсальность  
С. простота эксплуатации  
D. наличие поддержки (сопровождения) со стороны разработчика  
E. все ответы верны
5. Какие программы можно отнести к системному ПО  
А. утилиты  
В. экономические программы  
С. статистические программы  
D. мультимедийные программы  
E. все ответы верны
6. Этап, занимающий наибольшее время, при разработке программы  
А. тестирование  
В. сопровождение  
С. проектирование  
D. программирование  
E. формулировка требований
7. Первый этап в жизненном цикле программы  
А. формулирование требований  
В. анализ требований  
С. проектирование  
D. автономное тестирование  
E. комплексное тестирование
8. Один из необязательных этапов жизненного цикла программы



- A. оптимизация  
B. проектирование  
C. тестирование  
D. программирование  
E. анализ требований
9. Самый большой этап в жизненном цикле программы  
A. эксплуатация  
B. изучение предметной области  
C. программирование  
D. тестирование  
E. корректировка ошибок
10. Какой этап выполняется раньше  
A. отладка  
B. оптимизация  
C. программирование  
D. тестирование  
E. все ответы верны
11. Что выполняется раньше  
A. компиляция  
B. отладка  
C. компоновка  
D. тестирование  
E. нет правильного ответа
12. Что выполняется раньше  
A. проектирование  
B. программирование  
C. отладка  
D. тестирование  
E. компоновка
13. В стадии разработки программы не входит  
A. автоматизация программирования  
B. постановка задачи  
C. составление спецификаций  
D. эскизный проект  
E. тестирование
14. Самый важный критерий качества программы  
A. работоспособность  
B. надежность  
C. эффективность  
D. быстродействие  
E. простота эксплуатации
15. Способы оценки качества  
A. сравнение с аналогами  
B. наличие документации  
C. оптимизация программы  
D. структурирование алгоритма  
E. хранение и запоминание информации
16. Наиболее важный критерий качества  
17. надежность  
A. быстродействие  
B. удобство в эксплуатации  
C. удобный интерфейс  
D. эффективность
18. Способы оценки надежности  
A. тестирование  
B. сравнение с аналогами  
C. трассировка  
D. оптимизация  
E. удобный интерфейс

19. В каких единицах можно измерить надежность
- A. отказов/час
  - B. км/час
  - C. Кбайт/сек
  - D. операций/сек
  - E. мб/сек
20. В каких единицах можно измерить быстродействие
- A. отказов/час
  - B. км/час
  - C. Кбайт/сек
  - D. операций/сек
  - E. мб/сек
21. Что относится к этапу программирования
- A. написание кода программы
  - B. разработка интерфейса
  - C. работоспособность
  - D. анализ требований
  - E. создание базы данных
22. Последовательность этапов программирования
- A. компилирование, компоновка, отладка
  - B. компоновка, отладка, компилирование
  - C. отладка, компилирование, компоновка
  - D. компилирование, отладка, компоновка
  - E. все ответы верны
23. Инструментальные средства программирования
- A. компиляторы, интерпретаторы
  - B. СУБД (системы управления базами данных)
  - C. BIOS (базовая система ввода-вывода)
  - D. ОС (операционные системы)
  - E. нет правильного ответа
24. На языке программирования составляется
- A. исходный код
  - B. исполняемый код
  - C. объектный код
  - D. алгоритм
  - E. предметный код
25. Правила, которым должна следовать программа это
- A. алгоритм
  - B. структура
  - C. спецификация
  - D. состав информации
  - E. последовательность
26. Доступ, при котором записи файла читаются в физической последовательности, называется
- A. последовательным
  - B. прямым
  - C. простым
  - D. основным
  - E. вторичным
27. Доступ, при котором записи файла обрабатываются в произвольной последовательности, называется
- A. прямым
  - B. последовательным
  - C. простым
  - D. основным
  - E. вторичным
28. Методы программирования (укажите НЕ верный ответ)
- A. логическое
  - B. структурное
  - C. модульное

D.	компиляторное
E.	линейное
29.	Что выполняется раньше
A.	разработка алгоритма
B.	выбор языка программирования
C.	написание исходного кода
D.	компиляция
E.	Все ответы верны
30.	Найдите НЕ правильное условие для создания имен
A.	имена могут содержать пробелы
B.	длинное имя можно сократить
C.	из имени лучше выбрасывать гласные
D.	можно использовать большие буквы
E.	нет правильного ответа

### 5.2. Темы письменных работ

Формой осуществления контроля выполнения самостоятельной работы является подготовки рефератов на актуальные темы, т. е. изучение с помощью научных методов явлений и процессов, анализа влияния на них различных факторов, а также, изучение взаимодействия между явлениями, с целью получения убедительно доказанных и полезных для науки и практики решений с максимальным эффектом.

Цель реферата – определение конкретного объекта и всестороннее, достоверное изучение его структуры, характеристик, связей на основе разработанных в науке принципов и методов познания, а также получение полезных для деятельности человека результатов, внедрение в производство с дальнейшим эффектом.

Основой разработки каждой темы является методология, т. е. совокупность методов, способов, приемов и их определенная последовательность, принятая при разработке научного исследования. В конечном счете, методология – это схема, план решения поставленной научно-исследовательской задачи.

Процесс подготовки реферат состоит из следующих основных этапов:

1. Выбор темы и обоснование ее актуальности.
2. Составление библиографии, ознакомление с законодательными актами, нормативными документами и другими источниками, относящимися к теме проекта (работы).
3. Разработка алгоритма исследования, формирование требований к исходным данным, выбор методов и инструментальных средств анализа.
4. Сбор фактического материала.
5. Обработка и анализ полученной информации с применением современных методов анализа.
6. Формулировка выводов и выработка рекомендаций.
7. Оформление работы в соответствии с установленными требованиями.

Тематики рефератов по курсу “Технологии программирования”

1. Алгоритмизация. Понятие алгоритма и алгоритмической системы. Свойства алгоритма. Проектирование алгоритмов. Блок-схема алгоритма. Основные типы алгоритмов, их сложность и их использование для решения задач.
2. Основные принципы алгоритмизации и программирования. Понятие формализации, алгоритмизации, программирования.
3. Программа на языке высокого уровня, типы данных, переменные, выражения. Операторы циклов и ветвления.
4. Понятие о структурном программировании.
5. Объектно-ориентированное программирование.
6. Интегрированные среды программирования.
7. Этапы разработки программного обеспечения.
8. Основные понятия языков программирования. Трансляция. Компиляция и интерпретация.
9. Структуры и типы данных языка программирования.
10. Эволюция и классификация языков программирования

### 5.3. Фонд оценочных средств

1. Набор правил, методик и инструментов, позволяющих наладить производственный процесс выпуска какого-либо программного продукта - это:  
жизненный цикл программы  
технология программирования  
стандартизация программирования

2. Этапы (фазы) разработки, сопровождения программного продукта - это:  
жизненный цикл программы  
технология программирования  
стандартизация программирования

3. Начальный этап жизненного цикла программы - это:  
планирование  
оценка осуществимости

определение стоимости

4. Выберите верные утверждения:

спиральная модель ЖЦП допускает корректировки во время разработки  
постановка задачи - один из этапов ЖЦП  
необходимо документировать ход выполнения работы

5. Выберите верные утверждения:

планирование одна из фаз оценки осуществимости  
оценка осуществимости позволяет определить стоимость работы  
для оценки осуществимости необходима декомпозиция системы

6. Водопадная модель ЖЦП:

серия последовательных итераций при разработке программы  
требует единообразной оценки результатов на каждом этапе  
допускает корректировки по ходу разработки

7. Спиральная модель ЖЦП

серия последовательных итераций при разработке программы  
основывается на разработке работающего прототипа системы  
допускает корректировки по ходу разработки

8. Модель создания прототипов

серия последовательных итераций при разработке программы  
основывается на разработке работающего прототипа системы  
допускает корректировки по ходу разработки

9. Какой этап жизненного цикла программы выполняется вначале?

оценка осуществимости проекта  
техническое задание  
определение стоимости

10. Для правильной оценки осуществимости выполнения задачи необходимо:

декомпозиция системы на объекты  
анализ входных сигналов  
анализ выходных сигналов

11. Техническое задание:

разрабатывается заказчиком  
разрабатывается производителем  
разрабатывается заказчиком при согласовании с производителем

12. Документирование производственного процесса:

позволяет осуществить безболезненную замену одного из разработчиков при необходимости  
позволяет вычислить производительность каждого разработчика в коллективе  
увеличивает время на разработку

13. Оформление постановки задачи по стандартным правилам:

уменьшает вероятность разночтения пунктов задания разработчиками и заказчиком  
является ресурсоемкой операцией  
является одним из этапов жизненного цикла программы

14. Технология программирования зависит:

от инструментальных средств разработки  
от коллектива разработчиков  
от целевой ЭВМ

#### 5.4. Перечень видов оценочных средств

Задания со свободно конструируемым ответом (СКО) предполагает составление развернутого ответа на теоретический вопрос. Задание с выбором одного варианта ответа (ОВ, в задании данного типа предлагается несколько вариантов ответа, среди которых один верный. Задания со свободно конструируемым ответом (СКО) предполагает составление развернутого ответа, включающего полное решение задачи с пояснениями.

## 6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

### 6.1. Рекомендуемая литература

#### 6.1.1. Основная литература

Авторы, составители	Заглавие	Издательство, год
---------------------	----------	-------------------

	Авторы, составители	Заглавие	Издательство, год
Л1.1	Горюшкин А. А.	Офисное программное обеспечение: Учебное пособие	Москва: Русайнс, 2021, URL: <a href="https://book.ru/book/941651">https://book.ru/book/941651</a>
Л1.2	Точилкина Т. Е.	Моделирование бизнес-процессов. Практикум: Учебное пособие	Москва: КноРус, 2021, URL: <a href="https://book.ru/book/939105">https://book.ru/book/939105</a>
Л1.3	Иванова Г. С.	Технология программирования: Учебник	Москва: КноРус, 2021, URL: <a href="https://book.ru/book/938301">https://book.ru/book/938301</a>
Л1.4	Хорев П. Б.	Объектно-ориентированное программирование с примерами на C#: Учебное пособие	Москва: Издательство "ФОРУМ", 2020, URL: <a href="http://znanium.com/catalog/document?id=351782">http://znanium.com/catalog/document?id=351782</a>
Л1.5	Шакин В. Н., Загвоздкина А.В., Сосновиков Г. К.	Объектно-ориентированное программирование на Visual Basic в среде Visual Studio .NET: Учебное пособие	Москва: Издательство "ФОРУМ", 2020, URL: <a href="https://znanium.com/catalog/document?id=354206">https://znanium.com/catalog/document?id=354206</a>
Л1.6	Назаров С. В.	Архитектура и проектирование программных систем: Монография	Москва: ООО "Научно-издательский центр ИНФРА-М", 2020, URL: <a href="http://znanium.com/catalog/document?id=358698">http://znanium.com/catalog/document?id=358698</a>
Л1.7	Голицына О. Л., Партыка Т. Л.	Программное обеспечение: Учебное пособие	Москва: Издательство "ФОРУМ", 2019, URL: <a href="http://znanium.com/catalog/document?id=359201">http://znanium.com/catalog/document?id=359201</a>
Л1.8	Хорев П. Б.	Объектно-ориентированное программирование с примерами на C#: Учебное пособие	Москва: Издательство "ФОРУМ", 2021, URL: <a href="http://znanium.com/catalog/document?id=365883">http://znanium.com/catalog/document?id=365883</a>
Л1.9	Дадян Э.Г.	Современные технологии программирования. Язык C#: В 2 томах. Том 2	Москва: ООО "Научно-издательский центр ИНФРА-М", 2021, URL: <a href="https://znanium.com/catalog/document?id=379204">https://znanium.com/catalog/document?id=379204</a>

#### 6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год
Л2.1	Иванова Г. С.	Технология программирования: Учебник	Москва: КноРус, 2016, URL: <a href="https://book.ru/book/918438">https://book.ru/book/918438</a>
Л2.2	Костиков Ю.А., Мокряков А.В.	Особенности языка C# 5.0 .NET FRAMEWORK 4.5: Учебно-методическая литература	Москва: ООО "Научно-издательский центр ИНФРА-М", 2015, URL: <a href="http://znanium.com/catalog/document?id=150647">http://znanium.com/catalog/document?id=150647</a>
Л2.3	Баранова И. В., Баранов С.Н.	Объектно-ориентированное программирование на C++: Учебник	Красноярск: Сибирский федеральный университет, 2019, URL: <a href="http://znanium.com/catalog/document?id=380554">http://znanium.com/catalog/document?id=380554</a>

#### 6.2. Электронные учебные издания и электронные образовательные ресурсы

Э1	Интернет университет информационных технологий. - Режим доступа: <a href="https://www.intuit.ru/studies/courses%20">https://www.intuit.ru/studies/courses%20</a>		
Э2	Естественно-научный образовательный портал. - Режим доступа: <a href="http://www.en.edu.ru/">http://www.en.edu.ru/</a>		
Э3	Электронная библиотечная система Znanium. - Режим доступа: <a href="http://znanium.com/%20">http://znanium.com/%20</a>		
Э4	Электронные ресурсы Академии ИМСИТ. - Режим доступа: <a href="http://eios.imsit.ru/">http://eios.imsit.ru/</a>		
Э5	Электронная библиотечная система Ibooks. - Режим доступа: <a href="http://www.ibooks.ru/">http://www.ibooks.ru/</a>		
Э6	Электронная библиотечная система BOOK.ru. - Режим доступа: <a href="http://rpd.eios.imsit.ru:8080/RPD/Index/1636711/%20http://www.book.ru">http://rpd.eios.imsit.ru:8080/RPD/Index/1636711/%20http://www.book.ru</a>		

#### 6.3.1. Лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства

6.3.1.1	Windows 10 Pro RUS Операционная система – Windows 10 Pro RUS Подписка Microsoft Imagine Premium – Order №143659 от 12.07.2021		
6.3.1.2	Яндекс Браузер Браузер Яндекс Браузер Лицензионное соглашение на использование программ Яндекс Браузер <a href="https://yandex.ru/legal/browser_agreement/">https://yandex.ru/legal/browser_agreement/</a>		
6.3.1.3	Mozilla Firefox Браузер Mozilla Firefox Программное обеспечение по лицензии GNU GPL		
6.3.1.4	LibreOffice Офисный пакет LibreOffice Программное обеспечение по лицензии GNU GPL		
6.3.1.5	MS Visio Pro 2016 Интегрированная среда разработки Microsoft Visio профессиональный 2016 Подписка Microsoft Imagine Premium – Order №143659 от 12.07.2021		
6.3.1.6	MS Visual Studio Community Edition Среда разработки Microsoft Visual Studio 2022 Программное обеспечение по лицензии GNU GPL		

#### 6.3.2. Перечень профессиональных баз данных и информационных справочных систем

6.3.2.1	ABOUT THE UNIFIED MODELING LANGUAGE SPECIFICATION <a href="https://www.omg.org/spec/UML">https://www.omg.org/spec/UML</a>
6.3.2.2	ИСО Международная организация по стандартизации <a href="https://www.iso.org/ru/home.html">https://www.iso.org/ru/home.html</a>
6.3.2.3	РОССТАНДАРТ Федеральное агентство по техническому регулированию и метрологии <a href="https://www.gost.ru/portal/gost/">https://www.gost.ru/portal/gost/</a>
6.3.2.4	Кодекс – Профессиональные справочные системы <a href="https://kodeks.ru">https://kodeks.ru</a>

7. МТО (оборудование и технические средства обучения)			
Ауд	Наименование	ПО	Оснащение
120	Кабинет информатики, технологий и методов программирования	Windows 10 Pro RUS 7-Zip Яндекс Браузер Mozilla Firefox LibreOffice LibreCAD Inkscape Notepad++. 1С:Предприятие 8. Комплект Kaspersky Endpoint Security MS Access 2016 MS Project Pro 2016 MS SQL Server 2019 MS SQL Server Management Studio 18.8 MS Visio Pro 2016 MS Visual Studio Community Edition Visual Studio Code Blender Gimp Maxima Oracle VM VirtualBox StarUML V1 PostgreSQL IntelliJ IDEA PyCharm Community Edition Eclips Adobe Reader DC Arduino Software (IDE) NetBeans IDE ZEAL ARIS Express Archimate Klite Mega Codec Pack УМКК "Объектно-ориентированные технологии" УМКК "Основы алгоритмизации и программирования" Ramus Educational Micro-Cap Evaluation gvSIG Desktop Python	Стол - 20 шт., стул - 21 шт., рабочее место преподавателя – 1 шт., персональный компьютер с выходом в интернет - 21 шт., доска учебная – 1 шт., многофункциональное устройство – 1 шт., мультимедийный проектор – 1 шт., проекционный экран – 1 шт., интерактивная панель EliteBoard LR-75UT40i7 - 1 шт., соответствующее программное обеспечение
121	Лаборатория технологий программирования	Windows 10 Pro RUS 7-Zip Яндекс Браузер Mozilla Firefox LibreOffice LibreCAD Inkscape Notepad++. 1С:Предприятие 8. Комплект Kaspersky Endpoint Security MS Access 2016 MS Project Pro 2016 MS SQL Server 2019 MS SQL Server Management Studio 18.8	Стол - 20 шт., стул - 21 шт., рабочее место преподавателя – 1 шт., персональный компьютер с выходом в интернет - 21 шт., доска учебная – 1 шт., многофункциональное устройство – 1 шт., мультимедийный проектор – 1 шт., проекционный экран – 1 шт., соответствующее программное обеспечение

		MS Visio Pro 2016 MS Visual Studio Community Edition Visual Studio Code Blender Gimp Maxima Oracle VM VirtualBox StarUML V1 PostgreSQL IntelliJ IDEA PyCharm Community Edition Eclips Adobe Reader DC Arduino Software (IDE) NetBeans IDE ZEAL ARIS Express Archimate ПО ЛИНКО v8.2 демо-версия Klite Mega Codec Pack MS Office Standart 2007 Ramus Educational Micro-Cap Evaluation gvSIG Desktop Python	
Читальный зал	Информационно-библиотечный центр (помещение для самостоятельной работы обучающихся)	7-Zip Яндекс Браузер Mozilla Firefox LibreOffice LibreCAD Inkscape Notepad++. Kaspersky Endpoint Security MS Access 2016 MS Project Pro 2016 MS Visio Pro 2016 Visual Studio Code Blender Gimp Maxima IntelliJ IDEA PyCharm Community Edition Adobe Reader DC MS Office Standart 2007 Windows 10 Pro	Стол - 20 шт., стул - 20 шт., рабочее место сотрудника - 2 шт., персональный компьютер с выходом в интернет и обеспечением доступа в электронную информационно-образовательную среду академии – 17 шт., многофункциональное устройство – 2 шт.

### 8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

В соответствии с требованиями ФГОС ВО по направлению подготовки реализация компетентностного подхода предусматривает использование в учебном процессе активных и интерактивных форм проведения занятий (разбор конкретных задач, проведение блиц-опросов, исследовательские работы) в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

Лекционные занятия дополняются ПЗ и различными формами СРС с учебной и научной литературой. В процессе такой работы студенты приобретают навыки «глубокого чтения» - анализа и интерпретации текстов по методологии и методике дисциплины.

Учебный материал по дисциплине «Технологии программирования». разделен на логически завершённые части (модули), после изучения, которых предусматривается аттестация в форме письменных тестов, контрольных работ.

Работы оцениваются в баллах, сумма которых дает рейтинг каждого обучающегося. В баллах оцениваются не только знания и навыки обучающихся, но и их творческие возможности: активность, неординарность решений поставленных проблем. Каждый модуль учебной дисциплины включает обязательные виды работ – лекции, ПЗ, различные виды СРС (выполнение домашних заданий по решению задач, подготовка к лекциям и практическим занятиям).

Форма текущего контроля знаний – работа студента на практическом занятии, опрос. Форма промежуточных аттестаций – контрольная работа в аудитории, домашняя работа. Итоговая форма контроля знаний по модулям – контрольная работа с задачами по материалу модуля.

Методические указания по выполнению всех видов учебной работы размещены в электронной образовательной среде академии.

Методические указания и материалы по видам учебных занятий по дисциплине:

Вид учебных занятий, работ - Организация деятельности обучающегося

Лекция - Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения, отмечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, попытаться найти ответ в рекомендуемой литературе, если самостоятельно не удаётся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии.

Практические занятия - Конспектирование источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы, работа с текстом. Выполнение практических задач в инструментальных средах. Выполнение проектов. Решение расчётно-графических заданий, решение задач по алгоритму и др.

Самостоятельная работа - Знакомство с основной и дополнительной литературой, включая справочные издания, зарубежные источники, конспект основных положений, терминов, сведений, требующихся для запоминания и являющихся основополагающими в этой теме. Составление аннотаций к прочитанным литературным источникам и др.

## 9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ОБУЧАЮЩИМСЯ ПО ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа студентов в ходе семестра является важной составной частью учебного процесса и необходима для закрепления и углубления знаний, полученных в период сессии на лекциях, практических и интерактивных занятиях, а также для индивидуального изучения дисциплины «Технологии программирования» в соответствии с программой и рекомендованной литературой.

Самостоятельная работа выполняется в виде подготовки домашнего задания или сообщения по отдельным вопросам, написание и защита научно-исследовательского проекта.

Контроль качества выполнения самостоятельной (домашней) работы может осуществляться с помощью устного опроса на лекциях или практических занятиях, обсуждения подготовленных научно-исследовательских проектов, проведения тестирования.

Устные формы контроля помогут оценить владение студентами жанрами научной речи (дискуссия, диспут, сообщение, доклад и др.), в которых раскрывается умение студентов передать нужную информацию, грамотно использовать языковые средства, а также ораторские приемы для контакта с аудиторией.

Письменные работы позволяют оценить владение источниками, научным стилем изложения, для которого характерны: логичность, точность терминологии, обобщенность и отвлеченность, насыщенность фактической информацией.

Выбор варианта задания может быть сделан из предложенного ниже списка:

Вариант 1. Разработать игру «Мишень». Сценарий предлагает игроку указать координаты точки внутри квадрата с ребром  $a$ . Квадрат содержит круглую мишень, разбитую на кольца, пронумерованные от внешнего края к центру цифрами 1,2,3,...,9,10. Если игрок указал координаты точки внутри мишени, то сценарий возвращает номер кольца, куда «попала» точка, если он не попал по мишени, то сценарий возвращает 0. Каждый игрок получает  $n$  попыток. Сценарий суммирует номера колец, куда попал игрок. Победителем признаётся тот игрок, который набрал больше всего очков. Координаты точки должны вводиться, а результаты стрельбы должны отображаться с помощью диалоговых окон. Баллы за одно попадание должны начисляться с помощью приложения.

Вариант 2. Имеются две плоских фигуры: квадрат и круг. Причём площадь круга в три раза меньше, чем площадь квадрата. Разработать оконное приложение, который вначале требует ввести радиус и координаты центра круга, а затем сообщает какую площадь от площади квадрата в процентах занял круг с введёнными параметрами.

Вариант 3. Имеется плоская фигура треугольник. Разработать оконное приложение, которое реализует ввод координат вершин фигуры. При вводе угла поворота в градусах, приложение должно пересчитать координаты вершин повернутой фигуры. Вычисления должны вычисляться в ответ на изменения угла поворота, а не с помощью кнопки.

Вариант 4. Разработать оконное приложение, которое запрашивает у пользователя ввод строки с математическим выражением с одной переменной, имени переменной и её значения. Затем приложение вычисляет значение данного выражения, а результат выводит в диалоговое окно. Например, выражение: " $2*a/4-1$ ", имя переменной: " $a$ ", значение переменной: " $2$ ". Указание: воспользоваться методом `Replace` класса `String` языка `C#`.

Вариант 5. Разработать оконное приложение, который запрашивает у пользователя строку, и удаляет из неё все лишние пробелы. Считать, что строка не содержит лишних пробелов, если у неё отсутствуют пробелы в начале и в конце строки, а слова отделены только одним пробелом. Указание: воспользоваться методом `Split` класса `String` языка `C#`.

Вариант 6. Создать оконное приложение, которое должен запросить у пользователя его дату рождения. Если возраст менее 18, сценарий выводит диалоговое окно с сообщением «Привет, малыш!», если возраст более или равен 18 и менее 35, то «Здравствуйте, юноша!», если возраст не более 35, то «Здравствуйте!». Если при этом возраст совпадёт с вашим (вы одногодки), то «Привет!». Указание: воспользоваться методами класса `DateTime` (ДатаВремя).

Вариант 7. Создать оконное приложение, который определяет, является ли строка правильным скобочным выражением, т.е. для каждой открывающей скобки должна существовать закрывающая (закон Дика). Например, строка "`((()())`" – правильное скобочное выражение, а "`((()())`" – неправильное. Указание: для решения задачи можно воспользоваться следующим алгоритмом: если текущий символ строки равен '(' , то он добавляется к вершине стека, если он – ')', то из вершины стека символ удаляется. Если на выходе стек будет пустым, то скобочное выражение является правильным. Для



работы со стеком использовать методы Pop() и Push() класса System.Collections.Stack.

Вариант 8. Словом называется последовательность символов строки, исключая пробел, которая отделена от других символов пробелами. Создать оконное приложение, которое возвращает самое короткое слово строки. Считать, что пустая строка не является словом. Указание: Воспользоваться методами Split() и свойством Length класса String.

Вариант 9. Создать оконное приложение для двух массивов: countries – с названием стран, и population – с населением этих стран. Вывести название стран в форме таблицы (использовать визуальный компонент DataGridView (Сетка)). Реализовать вывод страны и её населения в ответ на щелчок мыши на соответствующей строке таблицы. Для вывода использовать компонент TextBox.

Вариант 10. Создать оконное приложение, содержащее список дат текущего месяца. В ответ на щелчок по дате в списке, приложение должно вывести в окно сообщения соответствующий день недели и текущее время. Указание: для работы с датами и временем использовать методы класса DateTime.

Вариант 11. Разработать оконное приложение, которое отображает список предстоящих телепередач, которые интересны пользователю. В ответ на щелчок по интересующей телепередаче, приложение должно сообщить сколько времени в часах и минутах осталось до её начала.

Вариант 12. Разработать оконное приложение, содержащее две таблицы (компонент DataGridView (Сетка)). Первая таблица содержит список названий и номеров кабинетов отделов организации, а вторая – список сотрудников с фамилиями, именами, должностями и номерами кабинетов, где находятся их рабочие места. В ответ на щелчок по строке с данными отделов, приложение должно обновить таблицу с данными сотрудников так, чтобы отображались только данные сотрудников, рабочие места которых находятся в кабинете выбранного отдела.

Вариант 13. Разработать оконное приложение, содержащее таблицу (компонент DataGridView (Сетка)), которая содержит список названий и номеров кабинетов отделов организации. Сразу после открытия главного окна приложения, строки таблицы должны быть заполнены данными из внешнего текстового файла.

Вариант 14. Разработать оконное приложение, содержащее две таблицы (компонент DataGridView (Сетка)). Первая таблица содержит список названий и номеров кабинетов отделов организации, а вторая – список сотрудников с фамилиями, именами, должностями и номерами кабинетов, где находятся их рабочие места. В ответ на нажатие кнопки, приложение должно удалить данные сотрудников, рабочие места которых находятся в кабинете выбранного отдела.

Вариант 15. Разработать оконное приложение, содержащее две таблицы (компонент DataGridView (Сетка)). Первая таблица содержит список названий и номеров кабинетов отделов организации, а вторая – список сотрудников с фамилиями, именами, должностями и номерами кабинетов, где находятся их рабочие места. В ответ на нажатие клавиши Enter, приложение должно отобразить ФИО сотрудников, рабочие места которых находятся в кабинете выбранного отдела, в окне MessageBox().

Вариант 16. Разработать оконное приложение, содержащее две таблицы (компонент DataGridView (Сетка)). Первая таблица содержит список названий и номеров кабинетов отделов организации, а вторая – список сотрудников с фамилиями, именами, должностями и номерами кабинетов, где находятся их рабочие места. В ответ на щелчок мышью по ячейке строки таблицы и именами сотрудников, приложение должно отобразить наименование соответствующего отдела в окне MessageBox().

Вариант 17. Разработать оконное приложение, содержащее таблицу (компонент DataGridView (Сетка)) со списком сотрудников с фамилиями, именами, должностями и номерами кабинетов, где находятся их рабочие места. В ответ на нажатие на кнопку, приложение должно сохранить во внешнем текстовом файле данные о сотруднике, выбранном в таблице.

Вариант 18. Разработать оконное приложение, содержащее поля ввода TextBox для названия отдела организации и номера его кабинета, а также таблицу (компонент DataGridView (Сетка)) со списком сотрудников с фамилиями, именами, должностями и номерами кабинетов, где находятся их рабочие места. В ответ на нажатие кнопки, приложение должно отобразить в таблице все данные только о тех сотрудниках, которые работают в отделе, чьи данные отображены в полях ввода.

Вариант 19. Разработать оконное приложение, содержащее выпадающий список (элемент ComboBox) названий отделов организации, а также таблицу (компонент DataGridView (Сетка)) со списком сотрудников с фамилиями, именами, должностями и номерами кабинетов, где находятся их рабочие места. В ответ на выбор названия отдела из списка, приложение должно отобразить все данные о сотрудниках, которые работают в отделе, чьи данные отображены в полях ввода.

Вариант 20. Разработать оконное приложение, содержащее список блюд кафе, оформленных в виде флажков (элемент CheckBox). Программа составляет список меню, состоящее из блюд, чьи флажки отмечены. Меню должно быть оформлено в виде документа в формате rtf (компонент RichTextBox) и может быть сохранено на внешнем носителе.