

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
имени И.Т. ТРУБИЛИНА»**

ФАКУЛЬТЕТ ПРИКЛАДНОЙ ИНФОРМАТИКИ



Рабочая программа дисциплины
Программирование

Направление подготовки
38.03.05 «Бизнес-информатика»

Направленность
«Анализ, моделирование и формирование интегрального представления стратегий и целей, бизнес-процессов и информационно-логической инфраструктуры предприятий и организаций»

Уровень высшего образования
Бакалавриат

Форма обучения
очная

Краснодар
2021

Рабочая программа дисциплины «Программирование» разработана на основе ФГОС ВО 38.03.05 Бизнес-информатика утвержденного приказом Министерства образования и науки РФ от 29 июля 2020 г. № 838.

Автор:

канд.пед.наук, доцент

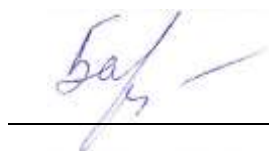


Т.А. Крамаренко

Рабочая программа обсуждена и рекомендована к утверждению решением кафедры системного анализа и обработки информации от 31.05.2021 г., протокол № 9а.

Заведующий кафедрой

д-р экон. наук, профессор



Т.П. Барановская

Рабочая программа одобрена на заседании методической комиссии факультета прикладной информатики, протокол от 31.05.2021 г. №9.

Председатель

методической комиссии

канд. пед. наук, доцент



Т.А. Крамаренко

Руководитель

основной профессиональной
образовательной программы

канд. экон. наук, доцент



А.Е. Вострокнутов

1 Цель и задачи освоения дисциплины

Целью освоения дисциплины «Программирование» является формирование комплекса знаний об организационных, научных и методических основах разработки алгоритмов и реализация их на одном из языков программирования высокого уровня с использованием процедурного подхода разработки программ.

Задачи

- приобретение системы знаний об алгоритмизации и программировании как одной из функций деятельности информатика, позволяющей принимать соответствующие решения при разработке программ;
- организация информационной системы для широкого круга внутренних и внешних пользователей;
- формирование и представление информации, удовлетворяющей требованиям различных пользователей программного обеспечения;
- изучение методов и способов разработки алгоритмов и программ с использованием языка высокого уровня C++;
- технические и программные средства реализации информационных процессов.

2 Перечень планируемых результатов по дисциплине, соотнесенных с планируемыми результатами освоения ОПОП ВО

В результате освоения дисциплины формируются следующие компетенции:

ОПК-3. Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации.

3 Место дисциплины в структуре ОПОП ВО

«Программирование» является дисциплиной обязательной части ОПОП ВО подготовки обучающихся по направлению 38.03.05 «Бизнес-информатика», направленность «Анализ, моделирование и формирование интегрального представления стратегий и целей, бизнес-процессов и информационно-логической инфраструктуры предприятий и организаций».

4 Объем дисциплины (252 часа, 7 зачетных единиц)

Виды учебной работы	Объем, часов	
	Очная	Заочная
Контактная работа в том числе:	162	
— аудиторная по видам учебных занятий	158	
— лекции	66	
— практические	-	
— лабораторные	92	
— внеаудиторная:	4	
— зачет	1	
— экзамен	3	
— защита курсовых проектов (работ)		
Самостоятельная работа в том числе:	90	
— курсовой проект (работа)		
— прочие виды самостоятельной работы	90	
Итого по дисциплине	252	

5 Содержание дисциплины

По итогам изучаемого курса студенты сдают зачёт (1 семестр) и экзамен (2 семестр).

Дисциплина изучается на 1 курсе, в 1 и 2 семестрах.

Содержание и структура дисциплины по очной форме обучения

№ п/п	Тема. Основные вопросы	Формируемые компетенции	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)			
				Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа
1	Алгоритмизация процессов обработки данных Цели и задачи курса. Основные понятия. Свойства алгоритмов. Способы записи алгоритмов. Запись алгоритмов в виде блок-схем. Этапы решения прикладных задач с использованием компьютера.	ОПК-3	1	2		4	8

№ п/п	Тема. Основные вопросы	Формируемые компетенции	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)			
				Лек ции	Практи- ческие занятия	Лабора- торные занятия	Самосто- ятельная работа
2	Введение в программирование Основные понятия программирования. История и принципы развития языков программирования. Типы языков программирования. Виды языков высокого уровня. Системы программирования	ОПК-3	1	2			4
3	Особенности разработки программы на C++. Лексические основы языка Общая характеристика языка. Технология разработки программ. Примерная структура исходного модуля программы в процедурно-ориентированном стиле. Препроцессор. Директивы препроцессора. Определение главной функции. Обзор сред программирования. Алфавит, особенности языка. Классификация типов данных. Операции, выражения и операторы. Объявление переменных. Задание констант. Операторы ввода-вывода. Область видимости и время жизни переменных. Преобразование типов. Библиотека math	ОПК-3	1	8		10	8
4	Программирование разветвлений на C++ Условие в программировании. Простые и составные условия. Условный оператор if(). Условная операция ? : в C++. Оператор множественного выбора switch() в C++. Особенности использования оператора break.	ОПК-3	1	4		4	4
5	Реализация алгоритмов с циклической структурой на C++ Понятие циклических вычислительных процессов. Цикл for(): синтаксис, особенности. Цикл с предусловием while(): синтаксис, особенности. Цикл с постусловием do(): син-	ОПК-3	1	4		12	8

№ п/п	Тема. Основные вопросы	Формируемые компетенции	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)			
				Лек ции	Практи- ческие занятия	Лабора- торные занятия	Самосто- ятельная работа
	таксис, особенности. Итерационные циклы. Особенности использования и способы задания условия в итерационном цикле. Вложенные циклы. Бесконечный цикл. Операторы continue и break в циклах. Функция exit().						
6	Массивы одномерные и двумерные в C++. Массив как тип данных. Определение и инициализация одномерного массива. Доступ к элементам. Определение и инициализация двумерного массива. Доступ к элементам. Обработка массивов	ОПК-3	1	4		4	6
7	Строки как массив символов Строка как массив символов типа char. Строковые переменные и константы: объявление и инициализация, вывод на экран. Одномерный массив строк. Операции со строками: копирование, сравнение, поиск, добавление строки к строке, определение длины, изменение регистра букв, кодировки букв и др.	ОПК-3	2	2		4	4
8	Сортировка массивов Понятие сортировки. Методы сортировки массивов. Алгоритм пузырьковой сортировки. Достоинства и недостатки метода. Реализация на C++ пузырьковой сортировки числовых массивов и строк.	ОПК-3	2	2		4	4
9	Структуры. Объединения. Перечисления Понятие структуры. Особенности структур. Определение структуры. Способы описания вложенных структур. Объявление структурных переменных. Использование операции доступа к полю структуры, в том чис-	ОПК-3	2	4		4	4

№ п/п	Тема. Основные вопросы	Формируемые компетенции	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)			
				Лек ции	Практи- ческие занятия	Лабора- торные занятия	Самосто- ятельная работа
	ле вложенной. Способы инициа- лизации полей структуры, в том числе вложенной. Объявление и инициализация массива струк- тур. Действия со структурными переменными. Объявление типа перечисление. Объявление переменной типа перечисление. Объявление типа объединение. Объявление переменной типа объединение. Присваивание объ- единений друг другу.						
10	Программирование функций Понятие, назначение, особенно- сти функций. Определение, объ- явление (прототипы) функций. Вызов функции пользователя. Способы описания функций в C++. Использование формаль- ных и фактических параметров (аргументов) при обмене инфор- мацией с функцией. Рекурсив- ные функции.	ОПК- 3	2	10		8	10
11	Указатели. Динамические пе- ременные Указатель как тип данных. Син- таксис и семантика типа дан- ных. Получение адреса пере- менной, доступ к значению пе- ременной по адресу. Адресная арифметика. Связь указателей с массивами, в том числе строка- ми, как массивами, функциями, структурами. Использование указателей на указатели. Свободная память. Действия с динамическими объектами: со- здание, уничтожение, присваи- вание значения. Динамические массивы: объявление, обработка.	ОПК- 3	2	10		12	10
12	Разработка приложений с ис- пользованием файлового вво- да-вывода Основные понятия. Текстовые и двоичные файлы. Файловый	ОПК- 3	2	4		12	10

№ п/п	Тема. Основные вопросы	Формируемые компетенции	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)			
				Лек ции	Практи- ческие занятия	Лабора- торные занятия	Самосто- ятельная работа
	ввод/вывод с помощью потоков. Дополнительные функции файлового ввода / вывода. Режимы файлов. Произвольный доступ к файлам.						
13	Основы разработки графических приложений Разработка проекта приложения на основе стандартов управления проектами ISO/IEC 12207. Структура проекта. Интегрированная среда разработки программ. Компонент форма –Form. Обработка события создания формы. Обработка событий «мыши». Обработка событий нажатия клавиш клавиатуры. Использование компонентов Label, textBox, comboBox и др. Функции перевода данных из одного типа в другой. Функции выдачи сообщений на экран. Функции ввода данных из диалоговых окон в программу. Работа с файлами с использованием функций компонент. Работа с датами. Использование средств обработки графической информации.	ОПК-3	2	8		14	12
Итого				66	-	92	90

6 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Методические указания (для самостоятельной работы)

1. Крамаренко Т. А. Программирование : практикум. Ч. 1 [Электронный ресурс] / Т. А. Крамаренко, Е. А. Иванова. – Краснодар : КубГАУ, 2020. – 171 с. – Режим доступа: https://edu.kubsau.ru/file.php/118/Praktikum_Programmirovanie_2020_chast1_593606_v1_.PDF

2. Крамаренко Т. А. Программирование : практикум. Ч. 2 [Электронный ресурс] / Т. А. Крамаренко, Е. А. Иванова. – Краснодар : КубГАУ, 2020.

– 87 с. – Режим доступа:
https://edu.kubsau.ru/file.php/118/Praktikum_Programmirovanie_2020_chast2_594210_v1_.PDF.

3. Программирование : метод. рекомендации по организации самостоятельной работы [Электронный ресурс] / сост. Т. А. Крамаренко, Е. А. Иванова. – Электрон. текстовые данные. – Краснодар : КубГАУ, 2020. – 28 с. – Режим доступа:
https://edu.kubsau.ru/file.php/118/Metodichka_po_SR_Programmirovanie_594203_v1_.PDF.

4. Программирование на СИ/СИ++. Методические указания к лабораторным работам по дисциплине «Программирование»: Часть 1 [Электронный ресурс] / А.Г. Мурлин, Н.В. Ефанова, Е.А. Иванова, О.М. Гончар. – Краснодар: КубГАУ, 2012. – 119 с. - Режим доступа:
https://edu.kubsau.ru/file.php/118/01_Programmirovanie_chast_1_.pdf

5. Программирование на СИ/СИ++. Методические указания к лабораторным работам по дисциплине «Программирование»: Часть 2 [Электронный ресурс] / А.Г. Мурлин, Н.В. Ефанова, Е.А. Иванова, О.М. Гончар. – Краснодар: КубГАУ, 2012. – 118 с. - Режим доступа:
https://edu.kubsau.ru/file.php/118/02_Programmirovanie_chast_2_.pdf

7 Фонд оценочных средств для проведения промежуточной аттестации

7.1 Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП ВО

Номер семестра (номер семестра соответствует этапу формирования компетенции)	Этапы формирования и проверки уровня сформированности компетенций по дисциплинам, практикам в процессе освоения ОПОП ВО
<i>ОПК-3 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации</i>	
1, 2	Программирование
2	Учебная практика: ознакомительная практика
3	Объектно-ориентированное программирование
8	Защита выпускной квалификационной работы, включая подготовку к защите и процедуру защиты

7.2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

Планируемые результаты освоения компетенции (индикаторы достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетворительно» минимальный не достигнут	«удовлетворительно» минимальный (пороговый)	«хорошо» средний	«отлично» высокий	
ОПК-3. Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации					
ОПК-3.1 Правильно использует методы формализованного описания решения поставленной задачи	Уровень знаний ниже минимальных требований, имели место грубые ошибки При решении стандартных задач не продемонстрированы основные умения, имели место грубые ошибки	Минимально допустимый уровень знаний, допущено много негрубых ошибок Продемонстрированы основные умения, решены типовые задачи с негрубыми ошибками, выполнены все задания, но не в полном объеме	Уровень знаний в объеме, соответствующем программе подготовки, допущено несколько негрубых ошибок Продемонстрированы все основные умения, решены все основные задачи с негрубыми ошибками, выполнены все задания в полном объеме, но некоторые с недочетами	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок Продемонстрированы все основные умения, решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме	Реферат, задачи, тест, вопросы и задания для проведения зачета, вопросы и задачи для проведения экзамена
ОПК-3.2 Способен проводить реализацию алгоритма на языках программирования, в том числе объектно-ориентированных с использованием современных инструментальных средств разработки					
ОПК-3.4 Применяет на практике современные методы программирования, в том числе командной разработки, проводит сборку					

7.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков, характеризующих этапы формирования компетенций в процессе освоения ОПОП ВО

7.3.1 Оценочные средства по компетенциям ОПК-3 для текущего контроля

Лабораторные работы

Крамаренко Т. А. Программирование : практикум. Ч. 1 [Электронный ресурс] / Т. А. Крамаренко, Е. А. Иванова. – Краснодар : КубГАУ, 2020. – 171 с. – Режим доступа: https://edu.kubsau.ru/file.php/118/Praktikum_Programmirovaniye_2020_chast1_593606_v1_.PDF.

Крамаренко Т. А. Программирование : практикум. Ч. 2 [Электронный ресурс] / Т. А. Крамаренко, Е. А. Иванова. – Краснодар : КубГАУ, 2020. – 87 с. – Режим доступа: https://edu.kubsau.ru/file.php/118/Praktikum_Programmirovaniye_2020_chast2_594210_v1_.PDF.

Примеры тем рефератов

- 1 Основные этапы разработки алгоритмов при разработке программных продуктов.
- 2 Конструирование пользовательских типов данных при разработке программ.
- 3 Программирование на языке C++ при разработке программ системного назначения.
- 4 Алгоритмы сортировки массивов, сравнительный анализ и выбор предпочтения.
- 5 Язык программирования C++ и эффективность программного кода.

Текст реферата должен содержать аргументированное изложение определенной темы. Реферат должен быть структурирован (по главам, разделам, параграфам) и включать разделы: введение, основная часть, заключение, список используемых источников. В зависимости от тематики реферата к нему могут быть оформлены приложения, содержащие документы, иллюстрации, таблицы, схемы и т. д.

Примеры задач, заданий

Задачи и задания к теме «Алгоритмизация процессов обработки данных. Введение в программирование»

Вариант 1

1. Математическая модель – это ...

2. Укажите порядок этапов решения прикладных задач с использованием ЭВМ (поставьте порядковый номер):

- ☐ Разработка алгоритма
- ☐ Анализ и исследование задачи, модели
- ☐ Постановка задачи
- ☐ Программирование
- ☐ Тестирование и отладка
- ☐ Сопровождение программы
- ☐ Анализ результатов решения задачи и уточнение в случае необходимости математической модели с повторным выполнением этапов 2-5.

3. Алгоритм должен отвечать определенным требованиям:

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____

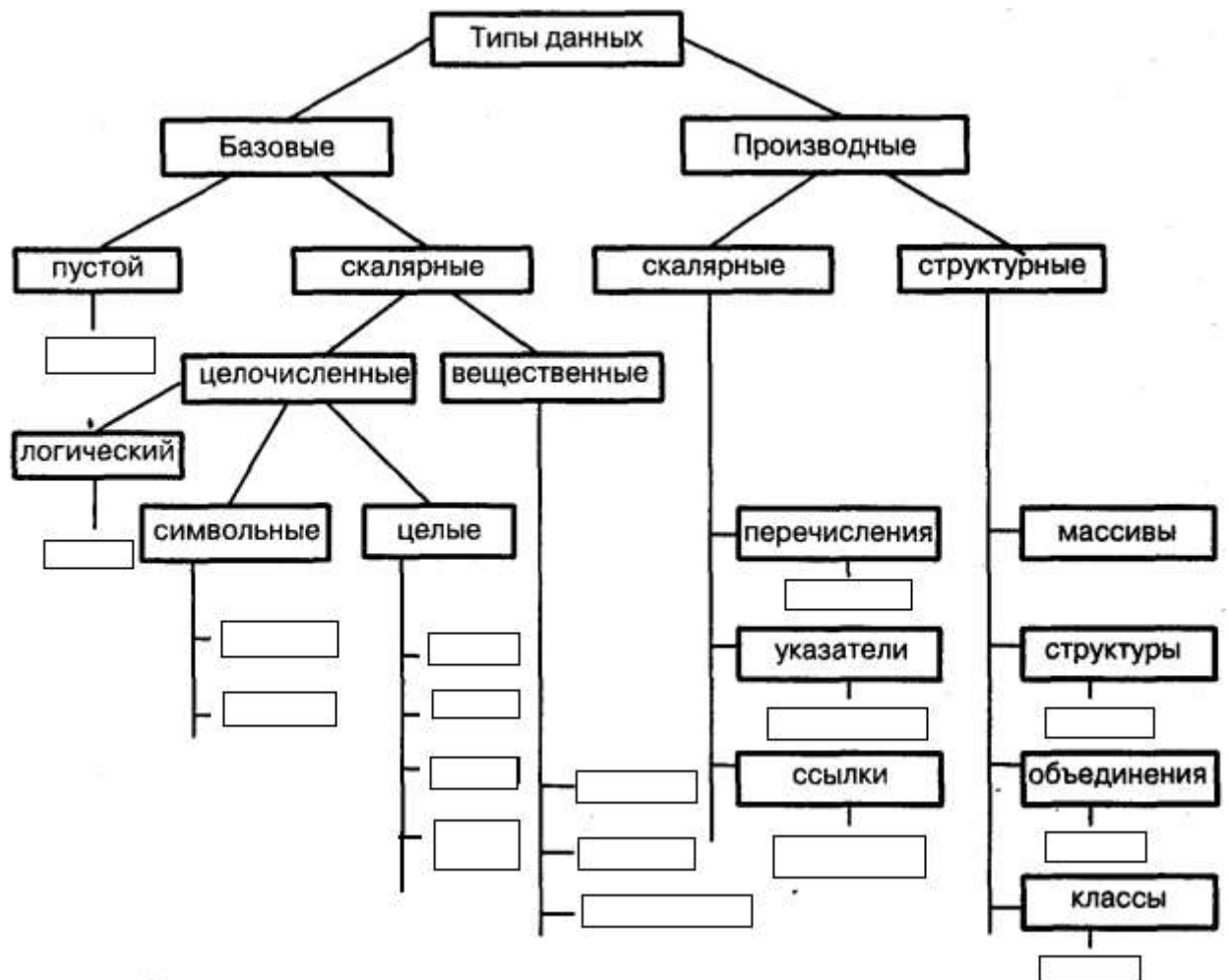
4. В алгоритме могут использоваться следующие базовые алгоритмические структуры:

5. Типы языков программирования

*Задачи и задания к теме «Особенности разработки программы на C++.
Лексические основы языка»*

Вариант №1

1. В дерево классификации типов языка C++ вписать имена соответствующих типов:



2. Указать ошибочные имена идентификаторов:

PROG_1, __PROG_1__, 1_Prog, prog#1, prog__1

3. Объявить двумя способами числовую вещественного типа символическую константу (именованную) PI со значением 3.1415:

1 сп.)

2 сп.)

4. Дан фрагмент программы. Определить, что будет выведено в результате его выполнения.

```
int x = 10;
```

```
int y = 3;
```

```
int z = x / y++;
```

```
cout<<"x = "<<x<<endl<<"y = "<<y<<endl<<"z = "<<z<<endl;
```

На экране:

5. Написать таблицу истинности для операции &&

6. В программе описаны следующие переменные:

```
int a=1, b=2, c=3;
```

Определить значение выражения

$a+b == c \ \&\& \ c-b < a$

7. Записать заданную формулу на языке C++: $\frac{x^{\cos x}}{|x + e^x| + \operatorname{tg} x}$

8. Даны переменные:

char c; int i; short s;

Определить тип результата смешанного выражения

Выражение	Тип
c - s / i	

9. Область видимости и время жизни переменных. Глобальные переменные. Где объявляются, примеры.

Задачи и задания к теме «Программирование разветвлений. Реализация алгоритмов с циклической структурой на C++»

Вариант 1

1. Дан следующий фрагмент программы?

```
...
int x,y,min;
cin>>x>>y;
if (x<y) min =x;
else min=y;
cout<<"min="<<min;
...
```

Записать фрагмент кода, выполняющий аналогичные действия с использованием операции условия ?:

2. Структура оператора множественного выбора (разветвление потока выполнения). Полная форма.

3. Особенности циклов всех типов.

4. В результате выполнения фрагмента программы

```
int a, n;
cin >> n;
for(a = 0; a < n; a++) cout << "Привет!";
```

При условии, что $n > 0$, на экране появится:

- Привет
- n раз слово Привет
- n-1 раз слово Привет
- Нет правильного ответа

5. Решить задачу: Даны три целых числа. Найти количество положительных чисел в исходном наборе.

6. Написать программу, содержащую операторы цикла, по указанию преподавателя (код написать полностью)

```

1
2  2
3  3  3
4  4  4  4
5  5  5  5  5

```

Задачи и задания к теме «Массивы одномерные и двумерные в C++. Строки как массив символов. Сортировка массивов»

Вариант 1

1. Что записано в следующих строках программы:

```
double mas[6]; //
int M[4][3]; //
```

2. Что будет выведено на экран в следующем фрагменте программы:

```
...
int A[3]={1, 2, 3};
...
cout<<A[0]+A[1]+A[2]<<"\n"; // На экране: _____
...
```

3. Найдите ошибку в объявлении и инициализации массива и напишите ниже правильную запись (если все верно, то напишите в области Ответ «ВЕРНО»):

```
int n[6]={7, 8, 3, 23, 32};
int m[2][3]={{7,8},{3,23},{3,2}};
```

Ответ:

```
int n[6]=
int m[2][3]=
```

4. Запишите объявление и ввод с клавиатуры элементов целочисленного массива `mas[m][n]`, если его размерность задается символическими константами `m=2` и `n=7`

5. Если `int A[4] = {1,3,5};`, то после выполнения

```
int c = ++A[2] + A[3]++;
c+=++A[3];
```

☐ `c = 7`

☐ `c = 8`

☐ `c = 9`

☐ `c = 16`

☐ `c = 17`

6. В заданном объявлении и инициализации строки с помощью массива символов определите размерность массива?

```
char str[]="ABCDE"; //
```

7. Написать команду конвертирования строки `str1` к нижнему регистру.
8. Определить результат сравнения следующих пар строк (проставить знак):
- ```
"123ab" "123ab"
"123ab" "123_ab"
"123AB" "123ab"
```
9. В программе описана строковая переменная `s`, значение которой равно:
- ```
"Hello, World!"
```
- Определить результат вызовов каждой из нижеперечисленных функций:

<code>strlen(s)</code>	
<code>strcmp(s, "Hello")</code>	
<code>strrchr(s, 'o')</code>	
<code>strstr(s, "or")</code>	
<code>strcat(s, " Hello")</code>	
<code>islower(s[7])</code>	

10. За сколько проходов будет отсортирован по возрастанию методом пузырьков следующий массив (указать позиции элементов массива в проходе):

54	-8	13	5	4

Задачи и задания к теме «Структуры. Программирование функций»

Вариант 1

1. Правильное определение структуры в C++

- `struct {int a;}`
- `struct a_struct {int a;};`
- `struct a struct int a;`
- `struct a_struct {int a;}`

2. Укажите правильный доступ к полю переменной типа структуры (не через указатель)

- `b->var;`
- `b>var;`
- `b.var;`
- `b-var;`

3. Выберите правильное (полное) определение функции.

o	<pre>int funct();</pre>	o	<pre>void funct(int)</pre>
o	<pre>{</pre>		<pre>{</pre>
	<pre>int funct(int x)</pre>		<pre> cout << "Hello"</pre>
	<pre>{</pre>		<pre>}</pre>
	<pre> return x = x + 1;</pre>	o	<pre>void funct(x)</pre>
	<pre>}</pre>		<pre>{</pre>
			<pre> cout << "Hello"</pre>
			<pre>}</pre>

4. Описана структура:

```
Struct predmet
{
    Char name[50];        //название предмета
    Int chasy;            //количество часов
    Bool ekzamen;         //есть ли экзамен
};
```

Найти ошибки в следующем фрагменте кода:

```
predmetp1 = {"Математика", "150", "true"};
p1.name = "Физика";
```

5. Описать структурный тип «Кинофильм». Поля: название, жанр, год выхода на экран. Создать переменную данного типа. При описании проинициализировать ее (произвольно). Описать массив из 5 кинофильмов (присваивать ему значения не нужно). Вывести на экран содержимое его 3-го элемента.

6. Написать программу, которая содержит вызов функции, устанавливающей, можно ли из отрезков с длинами a, b, c построить треугольник. Тип функции bool (1 - если треугольник существует, 0 - если нет).

Задачи и задания к теме «Указатели. Динамические переменные

Вариант 1

1. Перечисленные ниже определения являются определением

-
- переменная, содержащая адрес переменной;
 - переменная, содержащая адрес памяти, где расположены другие объекты (переменные, функции и т. п.);
 - символическое представление адреса.

2. Прокомментируйте каждую строку фрагмента программы

```
...
char cc='f', *pc;    // _____
                    // _____
int *p1, *p2;        // _____
pc=&cc;              // _____
int a=10, b=20, c=30; // _____
char *p=pc;          // _____
*p='+';              // _____
p1=&a;                // _____
p2=&b;                // _____
*p2=25;              // _____
```

```
cout<<" cc= "<<cc<<'\\n'; //На экране:_____
cout<<" a= "<<a<<'\\n'; //На экране:_____
cout<<" b= "<<b<<'\\n'; //На экране:_____
cout<<" c= "<<c<<'\\n'; //На экране:_____
cout<<" *p1= "<<*p1<<'\\n'; //На экране:_____
cout<<" *p2= "<<*p2<<'\\n'; //На экране:_____
cout<<" *pc= "<<*pc<<'\\n'; //На экране:_____
...

```

3. Что будет выведено на экран в следующем фрагменте программы:

...

```
char str[] = "Hello, world!";

char *b;

b = &str[7];

*b = 'W';

cout << b; // На экране:_____

```

4. Пусть есть одномерный массив целочисленного типа и указатель на переменную целого типа:

```
int a[5], *pa;
```

Напишите эквивалентную запись следующему присваиванию:

```
pa=&a[0]; // или _____
```

5. Напишите последовательность операторов, которая содержит:

- объявление целочисленного одномерного динамического массива размером n ;
- освобождение выделенной памяти.

Задачи и задания к теме «Разработка Windows-приложений с использованием файлового ввода-вывода»

Вариант 1

Разработайте в интегрированной среде программирования приложение Windows для решения двух задач с содержательной постановкой согласно варианту задания (первая задача – с линейным алгоритмом, вторая – с циклами). Организуйте решение каждой задачи на отдельной форме и создайте начальную форму для их вызова. Предусмотрите в приложении:

– *демонстрационный режим работы программы* (часть кнопок недоступна. После ввода правильного пароля режим меняется на полнофункциональный);

– *автоматическую проверку пароля*, сохраняемого в специальном файле. При запуске программы пароль считывается из этого файла и программа начинает работать в полнофункциональном режиме;

- *проверку корректности* вводимых исходных данных. При вводе неправильных данных (например, отрицательного количества) должно выводиться соответствующее сообщение и поле ввода должно очищаться;
- вывод текущего времени и/или времени работы программы (с использованием таймера);
- импорт-экспорт исходных данных с использованием текстовых файлов, вызываемых с помощью *стандартного окна выбора файла*;
- вывод *дополнительного диалогового окна*, например, со справочной информацией о программе и ее авторе;
- дублирование операций, запускаемых кнопками, с помощью *системы меню*.

№ варианта	Задача 1	Задача 2
1	Найдите корни квадратного уравнения x_1, x_2 ($a*x^2+b*x+c=0$)	Имеется набор замеров напряжения электрической сети. Определите среднее и максимальное напряжение в сети

Задачи и задания к теме «Разработка Windows-приложений с использованием графики»

Изобразить в окне приложения геометрическую фигуру согласно варианту. Реализовать с помощью таймера движение данной геометрической фигуры в заданном направлении.

№ варианта	Вид фигуры	Направление движения
1.		Влево

Примеры тестов

База тестовых заданий содержит в себе более 300 тестовых заданий по всем разделам дисциплины.

Тесты к теме «Алгоритмизация процессов обработки данных. Введение в программирование»

1... уровень не является уровнем языка программирования

- : Машинный
- : Машинно-ориентированный
- : Машинно-независимый
- +: Машинно-программный

2 Язык программирования C++ относится к ... языкам программирования

- : машинным
- : машинно-ориентированным
- +: машинно-независимым
- : графическим

3 Языки низкого уровня требуют ...

- : указания средних деталей процесса обработки данных
- +: указания мелких деталей процесса обработки данных
- : указания крупных деталей процесса обработки данных
- : описания алгоритмов

4... языки не относятся к языкам высокого уровня

- : Процедурные
- +: Машинно-ориентированные
- : Логические
- : Объектно-ориентированные

5 Язык программирования ... не относится к языкам высокого уровня

- : C++
- : Pascal
- +: Assembler
- : Basic

Тесты к теме «Реализация алгоритмов с циклической структурой на C++»

1 Найти значение c при int a=3,b=5,c=1:

```
while (a<b)    { c= c+a*b;  a= a+1; }
```

Ответ:

- | | |
|------------|-----------|
| 1 | 21 |
| 2 | 30 |
| 3 + | 31 |
| 4 | 32 |
| 5 | 41 |

2 Найти значение c при int a=10, b=5;

```
do {  b= b+1;  a= a-b;  c:= a+b; }  while( a>0);
```

Ответ:

- | | |
|------------|----------|
| 1 | 2 |
| 2 | 10 |
| 3 | 14 |
| 4 | 7 |
| 5 + | 4 |

3 Найти значение c для заданных int a=3,b=2,c=4,n=4;

```
for (inti=1;i<=n;i++) c:= c+a*b;
```

Ответ:

- | | |
|---|----|
| 1 | 18 |
|---|----|

2 38
3 8
4+ 28
5 100

4 Найти значение c

```
inta=2, b=4, c=3;  
while (a<b) {    c= c+a*b;    a= a+1; }
```

Ответ:

1+ 23
2 20
3 13
4 15
5 25

5 Найти значение c при

```
inta=9, b=4,c;  
do{ b= b+1;    a= a-b;    c= a+b;}  
while(a>0);
```

Ответ:

1 2
2 3
3 7
4 9
5+ 4

Тесты к теме «Массивы одномерные и двумерные в C++. Строки как массив символов»

1 Определить, что находит фрагмент программы:

```
intA[10],i,S;  
...  
S= A[0];    for (i= 1; i<10;i++)        if (S>A[i]) S=A [i];  
...
```

Ответ:

1 наибольший элемент массива
2 + наименьший элемент массива
3 среднее значение элементов массива
4 среднее квадратичное значение элементов массива
5 сумму элементов массива

2 Найти значение S:

```
int A[7],S=0;  
for (inti=0;i<= 6;i++) {    A [i]=i;    S+= A [i];}
```

Ответ:

1+	21
2	31
3	11
4	41
5	18

3 Найти значение S:

```

inti, A[5], S=1;
for (i=1; i<= 4; i++) { A [i]=i; S*= A [i]; }
S=(S+6)/i;

```

Ответ:

1	4
2	5
3+	6
4	7
5	8

4 Каким символом заканчивается символьная строка?

Ответ:

1+	'\0'
2	'0'
3	'\n'
4	'endl'
5	'\r'

5 Укажите правильный вариант копирования строк.

- a) charq [10] = "qwerty", s[10] = "01234";
strcpy (q, s);
- b) charq [10] = "qwerty", s[10] = "01234";
strcpy (s, q);
- c) char q [10] = "qwerty", s[10] = "01234";
strncpy (s+2, q);
- d) char q [10] = "qwerty", s[10] = "01234";
strncpy (q+1, s+2);
- e) char q [10] = "qwerty", s[10] = "01234";
strncpy (q+1, s+2, 2);

Ответ:

1	a)
2	b)
3	b),c)
4+ a),b)	
5	c),d),e)

1 Перечисляемый тип данных объявляется с помощью ключевого слова

....

- : struct
- : class
- +: enum
- : number
- : string

2 Структурой называется составной тип данных, состоящий из элементов ... типов.

- : одинаковых
- +: разных
- : перечисляемых
- : числовых
- : строковых

3 Для доступа к элементу структуры используется операция

- +: точка (.)
- : стрелка (→)
- : амперсанд (&)
- : процент (%)
- : решетка (#)

4 Основным отличием структур и объединений является то, что поля в объединениях всегда

- +: находятся в одной и той же области памяти
- : должны быть одного типа
- : должны быть одного размера
- : являются недоступными

5 Структуры в языке C++ отличаются от классов тем, что ...

- : они могут наследовать другие структуры
- : они могут использоваться в качестве базовых для других структур
- +: они относятся к типам значений
- : в них нельзя определять конструкторы

Тесты к теме «Основы разработки приложений с использованием инструментальной среды разработки и отладки программ в C++ Builder»

1 Основной принцип программирования под Windows:

- +: операционная система вызывает программу
- : программа вызывает операционную систему
- : программа и операционная система работают независимо друг от друга

-: программа встраивается в операционную систему

2В ОС Windows сообщения программе посылаются

- +: в случайные моменты времени
- : только в начале работы программы
- : только после обработки предыдущего сообщения
- : только в конце работы программы
- : периодически с заданным интервалом

3 Основные компоненты Windows-приложений:

- +: функция WinMain
- +: функция окна
- +: цикл обработки сообщений
- +: класс окна
- : главная форма программы
- : список поступивших программе сообщений
- : цикл перерисовки окна

4 GDI – это подсистема Windows, которая обеспечивает поддержку

- +: аппаратно-независимой графики
- : аппаратно-зависимой графики
- : векторной графики
- : удаленного доступа к базе данных
- : доступа к локальной базе данных

5 Понятие контекста устройства в графике используется для определения,

- +: на какое устройство осуществляется вывод
- : с какого устройства осуществляется чтение данных
- : какие устройства подключены к компьютеру
- : каким цветом выводить изображение

7.3.2 Оценочные средства для промежуточного контроля

7.3.2.1 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации

Компетенции: ОПК-3 - Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации

Вопросы к зачету

1. Понятие информационной, математической, компьютерной модели.
2. Этапы решения прикладных задач с использованием ЭВМ
3. Понятие алгоритма. Требования, предъявляемые к алгоритмам.
4. Базовые алгоритмические структуры (конструкции).
5. Понятие программы.
6. Структура языка программирования.
7. Понятие системы программирования. Транслятор.
8. Интегрированная среда программирования.
9. Языки программирования низкого уровня.
10. Виды языков программирования высокого уровня.
11. Этапы процесса создания программ на C++ (схема получения исполнимого модуля программы в интегрированной среде программирования).
12. Синтаксическая структура программы на языке C++: типовые части исходного модуля.
13. Определение главной функции.
14. Рекомендации по написанию синтаксически правильной программы.
15. Алфавит языка C++.
16. Особенности языка C++.
17. Типы данных в языке C++.
18. Понятие операции, выражения, оператора в C++.
19. Основные арифметические операции. Примеры.
20. Операции присваивания. Виды операции. Примеры.
21. Операции инкремента и декремента. Примеры.
22. Побитовые или булевские операции.
23. Объявление и инициализация переменных.
24. Константы. Способы объявления констант в C++.
25. Реализация ввода-вывода в C++.
26. Битовые поля.
27. Условие в программировании. Простые и составные условия.
28. Операции отношения (сравнения), логические операции.
29. Условный оператор.
30. Условная операция ? : в C++.
31. Оператор множественного выбора switch() в C++.
32. Особенности использования оператора break.
33. Цикл FOR: синтаксис, особенности.
34. Цикл с предусловием: синтаксис, особенности.
35. Цикл с постусловием: синтаксис, особенности.
36. Вложенные циклы.
37. Бесконечный цикл.
38. Операторы continue и break в циклах. Примеры.
39. Массив как тип данных. Определение и инициализация одномерного массива. Доступ к элементам.

40. Определение и инициализация двумерного массива. Доступ к элементам.
41. Строка как массив символов типа `char`.
42. Строковые переменные и константы: объявление и инициализация, вывод на экран.
43. Операции со строками: копирование, сравнение, поиск, добавление строки к строке, определение длины.
44. Сортировка одномерных массивов методом простого выбора.
45. Сортировка одномерных массивов методом пузырька.
46. Сортировка одномерных массивов методом вставок.

Компетенции: ОПК-3 - Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации

Вопросы к экзамену

1. Структура как тип данных. Синтаксис описания типа. Объявление структурных переменных.
2. Доступ к полям структуры. Инициализация полей структуры.
3. Массив структур.
4. Вложенные структуры.
5. Перечисление. Синтаксис описания типа. Объявление переменных.
6. Объединение. Синтаксис описания типа. Объявление переменных.
7. Основные понятия функций.
8. Особенности функций.
9. Определение функций.
10. Объявление (прототипы) функций.
11. Вызов функции.
12. Технология вставки функции пользователя в одном файле с основной программой.
13. Технология вставки функции пользователя в отдельный файл.
14. Библиотечные функции.
15. Использование формальных и фактических параметров (аргументов) при обмене информацией с функцией.
16. Передача констант и значений переменных в функцию
17. Передача аргументов по умолчанию (`default argument`) в функцию
18. Использование массивов в качестве параметров (аргументов) функции.
19. Структурные переменные в качестве аргументов функций.
20. Рекурсивные функции. Примеры.
21. Понятие ссылки. Синтаксис описания.
22. Основные понятия указателей.
23. Технология применения указателей.

24. Арифметические операции над указателями.
25. Указатели на массивы и их использование.
26. Указатели на функции и их использование.
27. Указатель на строку, объявленную как массив символов.
28. Указатели (массивы указателей) на строковые константы.
29. Определение одномерного массива строк через массивы указателей на строки.
30. Указатели на строки как аргументы функций.
31. Функции модуля `stdlib.h`, используемых для преобразования строковых данных в числовые.
32. Функции модуля `stdlib.h`, используемых для преобразования числовых данных в строковые.
33. Функции модуля `string.h`.
34. Массивы указателей на строки.
35. Указатели на структуры и их использование.
36. Указатель на `void` и его использование.
37. Технология использования указателей на указатели.
38. Свободная память. Операции управления `new` и `delete`.
39. Оператор `new` и `delete` при работе с простыми типами.
40. Оператор `new` и `delete` при работе с одномерными динамическими массивами.
41. Многомерные динамические массивы. Алгоритм выделения, алгоритм освобождения памяти.
42. Понятие файлов в C++. Текстовый формат записи в файл.
43. Понятие файлов в C++. Двоичный формат записи в файл. Особенности сохранения чисел в двоичном формате.
44. Запись информации в текстовый файл.
45. Чтение информации из текстового файла.
46. Режимы файлов.
47. Запись данных в двоичный файл.
48. Чтение данных из двоичного файла.
49. Дополнительные функции файлового ввода / вывода: функция `rdbuf()`.
50. Дополнительные функции файлового ввода / вывода: функции `write`(символьный массив, число символов), `read`(символьный массив, число символов), `gcount()`, `put`(параметр типа `char`).
51. Дополнительные функции файлового ввода / вывода: функция `get()`. Варианты функции `get()`.
52. Произвольный доступ к файлам: функции реализации «передвижения» по файлу в определенную точку файла. Пример реализации.
53. Произвольный доступ к файлам: функции для проверки текущей позиции файлового указателя.
54. Работа с файлом, как с базой данных. Пример реализации.
55. Копирование файлов. Пример реализации.

56. Основные понятия объектно-ориентированного программирования: объект, класс, поле, метод, свойства, сообщение, событие и т. д.
57. Визуальное программирование. Основные понятия.
58. Инструменты интегрированной среды MS Visual Studio 20XX.
59. Создание, открытие, сохранение, компиляция и запуск проекта на выполнение в интегрированной среде MS Visual Studio 20XX.
60. Компонент форма – TForm. Добавление к проекту, исключение из проекта, реализация связи между формами.
61. Компонент форма – TForm. Режимы запуска формы на выполнение.
62. Компонент форма – TForm. Свойства формы.
63. Компонент форма – TForm. Методы формы.
64. Компонент TLabel. Свойства компонента.
65. Компонент TEdit. Свойства, события и методы компонента.
66. Компонент TMemo.
67. Функции перевода данных из одного типа в другой в интегрированной среде MS Visual Studio 20XX. Примеры.
68. Функции выдачи сообщений на экран в интегрированной среде MS Visual Studio 20XX. Примеры.
69. Функции ввода данных из диалоговых окон в программу в интегрированной среде MS Visual Studio 20XX. Примеры.
70. Компоненты работы со списками в графическом приложении.
71. Компоненты, реализующие одиночный, множественный выбор в графическом приложении.
72. Компонент TMainMenu.
73. Компонент TPopupMenu.
74. Компоненты для работы с массивами данных в интегрированной среде MS Visual Studio 20XX.
75. Использование диалогов для выбора файлов в графическом приложении. Примеры.
76. Работа с файлами с использованием функций компонент в графическом приложении. Примеры.
77. Работа с датами в графическом приложении.
78. Построение графиков с помощью компонента TChart. Примеры.
79. Построение графиков с помощью свойства объекта TForm – Canvas. Примеры.

7.3.2.2 Практические задания для зачета

В рамках практического задания для оценки освоения компетенций ОПК-3 обучающемуся предлагается выполнить следующее задание.

Разработать консольное приложение согласно варианту. Предусмотреть проверку правильности ввода исходных данных.

№ вар.	Практическое задание
1	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и нахождения произведения сумм элементов массива с четными и нечетными номерами.
2	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и определения количества элементов, у которых значения предыдущего и последующего элемента равны? Например: 2 3 2 4 5 4 7 8 9 – таких элементов два, это 3 и 5.
3	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и определения суммы четных элементов с четными номерами, например: $V[2]=10$.
4	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и определения количества чисел, состоящих из одинаковых цифр? Например: 11, 55 и т.д.
5	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и подсчета произведения элементов массива V , значения которых совпадают с их номерами, например: $V[3]=3$.
6	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и изменения всех элементов массива на противоположные по знаку.
7	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива. Поменять местами минимальный и максимальный элементы массива.
8	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и определения суммы элементов, принадлежащих заданному диапазону.
9	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и определения количества нулей в этом массиве.
10	Разработать алгоритм и описать функции формирования элементов

	одномерного массива с помощью генератора случайных чисел и определения суммы его элементов, лежащих между его максимальным и минимальным элементами.
11	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и определения произведения его элементов, лежащих между его первым и последним нулями.
12	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и проверки равенства двух массивов.
13	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива. Вычислить сумму длин двух векторов. Длина вектора вычисляется по формуле: $ x = \sqrt{\sum x_i^2}$.
14	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и замены нечетных элементов на заданное число.
15	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и нахождения суммы и разности всех элементов с четными и нечетными номерами соответственно.
16	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и сортировки массива методом пузырька.
17	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и сдвига его элементов по правилу $1 \rightarrow 2 \rightarrow \dots n \rightarrow 1$.
18	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и нахождения суммы двух массивов.
19	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и определения того, образуют ли элементы массива неубывающую последовательность (функция должна возвращать «0», если все в порядке, или индекс первого элемента массива, на котором условие нарушается).
20	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и определения среднеарифметического значения элементов и макси-

	мального отклонения от него.
21	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и сортировки его таким образом, чтобы все положительные числа находились в начале, а отрицательные – в конце массива и был сохранен исходный порядок следования элементов в обеих группах.
22	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и нахождения пары соседних элементов, наиболее близко расположенных друг к другу. Мера близости: $R = A[i+1] - A[i] .$
23	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и нахождения суммы элементов массива, предшествующих первому отрицательному элементу. Если среди элементов массива нет отрицательных, то просуммировать все элементы.
24	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и определения, сколько раз в нем встретились два подряд идущих нулевых элемента.
25	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и поиска минимума среди тех элементов массива X , которые не являются элементами массива Y .
26	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и определения максимума из чисел, встречающихся в этом массиве более одного раза.
27	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и замены в нем каждой группы из подряд идущих нулей на 2 элемента: первый из них 0, а второй – количество нулей в данной группе.
28	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и сдвига элементов массива на k позиций вправо. Элементы, выходящие за границу массива, теряются. Освободившиеся в массиве пози-

	ции заполняются нулями.
29	Разработать алгоритм и описать функции ввода с клавиатуры элементов одномерного массива и замены в нем каждой группы из нечетного количества подряд идущих нулей на один ноль, а из четного количества – на два.
30	Разработать алгоритм и описать функции формирования элементов одномерного массива с помощью генератора случайных чисел и определения количества инверсий в этом массиве (т.е. таких пар элементов, в которых большее число находится слева от меньшего: $x_i > x_{i+1}$).

7.3.2.3 Практические задания для экзамена

В рамках практического задания для оценки освоения компетенций ОПК-3 обучающемуся предлагается выполнить следующее задание.

Разработать графическое приложение согласно варианту. Исходные данные вводить с помощью компонента TEdit в компонент TListBox либо с помощью свойства Text в свойство Items компонента TComboBox. Скалярный результат выводить с помощью компонента TLabel. Ввод строки заканчивать нажатием клавиши Enter. Для выхода из программы использовать кнопку Close. Для расчетов вводить несколько различных строк.

Варианты заданий:

1. Дана строка, состоящая из групп нулей и единиц. Каждая группа отделяется от другой одним или несколькими пробелами. Найти количество групп с пятью символами.
2. Дана строка, состоящая из групп нулей и единиц. Найти и вывести на экран самую короткую группу.
3. Дана строка, состоящая из групп нулей и единиц. Подсчитать количество символов в самой длинной группе.
4. Дана строка, состоящая из групп нулей и единиц. Найти и вывести на экран группы с четным количеством символов.
5. Дана строка символов, содержащая некоторый текст на русском языке. Разработать программу форматирования этого текста, т.е. его разбиения на отдельные строки (по k символов в каждой строке) и выравнивания по правой границе путем вставки между отдельными словами необходимого количества пробелов.
6. Дана строка, состоящая из букв, цифр, запятых, точек, знаков "+" и "-". Выделить подстроку, которая соответствует записи целого числа (т.е. начинается со знака "+" или "-" и внутри подстроки нет букв, запятых и точек).
7. Дана строка символов, состоящая из букв, цифр, запятых, точек, зна-

ков “+” и “-“. Выделить подстроку, которая соответствует записи вещественного числа с фиксированной точкой

8. Дана строка символов, состоящая из букв, цифр, запятых, точек, знаков “+” и “-“. Выделить подстроку, которая соответствует записи вещественного числа с плавающей точкой.

9. Дана строка символов, состоящая из произвольных десятичных цифр, разделенных пробелами. Вывести на экран числа этой строки в порядке возрастания их значений.

10. Дана строка символов, состоящая из произвольных десятичных цифр, разделенных пробелами. Вывести четные числа этой строки.

11. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Вывести на экран слова этого текста в порядке, соответствующем латинскому алфавиту.

12. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Вывести на экран порядковый номер слова, накрывающего k -ю позицию (если на k -ю позицию попадает пробел, то номер предыдущего слова).

13. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Разбить исходную строку на две подстроки, причем первая длиной k -символов (если на k -ю позицию попадает слово, то его следует отнести ко второй строке, дополнив первую пробелами до k -позиций).

14. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами.. Вывести на экран порядковый номер слова максимальной длины и номер позиции строки с которой оно начинается.

15. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Вывести на экран порядковый номер слова минимальной длины и количество символов в этом слове.

16. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. В каждом слове заменить первую букву на прописную.

17. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Удалить первые k слов из строки, сдвинув на их место последующие слова строки.

18. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Поменять местами i - и j -е слова.

19. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Поменять местами первую и последнюю буквы каждого слова.

20. Дана строка символов, состоящая из произвольного текста на английском языке, слова разделены пробелами. Заменить буквы латинского алфавита на соответствующие им буквы русского алфавита.

21. Дана строка символов $S_1S_2\dots S_m$, в которой могут встречаться циф-

ры, пробелы, буква «Е» и знаки “+”, “-“. Известно, что первый символ S1 является цифрой. Из данной строки выделить подстроки, разделенные пробелами. Определить, является ли первая подстрока числом. Если да, то выяснить: целое или вещественное число, положительное или отрицательное.

22. Дана строка, состоящая из групп нулей и единиц. Подсчитать количество единиц в группах с нечетным количеством символов.

23. Дана строка символов, содержащая некоторый текст на русском языке. Заменить буквы русского алфавита на соответствующие им буквы латинского алфавита.

24. Дана строка символов, содержащая некоторый текст. Разработать программу, которая определяет, является ли данный текст палиндромом, т.е. читается ли он слева направо так же, как и справа налево (например, «А роза упала на лапу Азора»).

25. Составить программу, которая читает построчно текст другой программы (ввести с клавиатуры) на языке C++, обнаруживает комментарии и выводит их на экран.

26. Составить программу, которая читает построчно текст другой программы (ввести с клавиатуры) на языке C++, подсчитывает количество ключевых пар символов «{» и «}» и выводит на экран соответствующее сообщение.

27. Разработать программу, которая заданное целое число от 1 до 1999 выводит на экран римскими цифрами.

28. Дан текст из заглавных латинских букв, за которым следует пробел. Определить, является ли этот текст правильной записью римскими цифрами целого числа от 1 до 999, и, если является, вывести на экран это число арабскими цифрами (в десятичной системе).

29. Дан текст из k символов. Вывести на экран только строчные русские буквы, входящие в этот текст.

30. Дан текст из k символов. Вывести на экран в алфавитном порядке все различные прописные русские буквы, входящие в этот текст.

7.4 Методические материалы, определяющие процедуры оценивания знаний, умений и навыков и опыта деятельности, характеризующих этапы формирования компетенций

Критериями оценки реферата являются: новизна текста, обоснованность выбора источников литературы, степень раскрытия сущности вопроса, соблюдения требований к оформлению.

Оценка «**отлично**» — выполнены все требования к написанию реферата: обозначена проблема и обоснована её актуальность; сделан анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция; сформулированы выводы, тема раскрыта полностью, выдержан объём; соблюдены требования к внешнему оформлению.

Оценка «**хорошо**» — основные требования к реферату выполнены, но

при этом допущены недочёты. В частности, имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; не выдержан объём реферата; имеются упущения в оформлении.

Оценка **«удовлетворительно»** — имеются существенные отступления от требований к реферированию. В частности: тема освещена лишь частично; допущены фактические ошибки в содержании реферата; отсутствуют выводы.

Оценка **«неудовлетворительно»** — тема реферата не раскрыта, обнаруживается существенное непонимание проблемы или реферат не представлен вовсе.

Критерии оценивания задач, заданий

Задача – средство, позволяющее оценить умение и навыки обучающегося применять стандартные методы решения поставленной задачи с использованием имеющейся инструментальной и (или) лабораторной базы, проводить анализ полученного результата работы.

Критерии оценки знаний студента при выполнении задачи.

Оценка **«отлично»** - выставляется обучающему, показавшему всесторонние, систематизированные, глубокие знания вопросов задания и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка **«хорошо»** - выставляется обучающему, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности, которые может устранить с помощью дополнительных вопросов преподавателя.

Оценка **«удовлетворительно»** - выставляется обучающемуся, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными понятиями выносимых на задачу тем, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации.

Оценка **«неудовлетворительно»** - выставляется обучающемуся, который не знает большей части основного содержания выносимых на задание вопросов тем дисциплины, допускает грубые ошибки в формулировках основных понятий и не умеет использовать полученные знания при решении типовых практических задач.

Критерии оценки при проведении тестирования

Оценка **«отлично»** выставляется при условии правильного ответа обучающегося не менее чем 85 % тестовых заданий;

Оценка **«хорошо»** выставляется при условии правильного ответа обучающегося не менее чем 70 % тестовых заданий;

Оценка **«удовлетворительно»** выставляется при условии правильного ответа обучающегося не менее 51 %;

Оценка «неудовлетворительно» выставляется при условии правильного ответа обучающегося менее чем на 50 % тестовых заданий.

Критерии оценки лабораторной работы

Оценка «**отлично**» выставляется в том случае, когда обучающийся правильно и полностью выполнил основное задание и, возможно при необходимости, дополнительное задание лабораторной работы, ответил правильно на теоретические вопросы, на дополнительные вопросы. Показал отличные знания и умения при выполнении лабораторной работы в рамках учебного материала.

Оценка «**хорошо**» выставляется в том случае, когда обучающийся правильно и полностью выполнил задание лабораторной работы, ответил на теоретические вопросы с небольшими неточностями, на большинство дополнительных вопросов также, возможно, допуская незначительные ошибки. Показал достаточно хорошие знания и умения при выполнении лабораторной работы в рамках учебного материала.

Оценка «**удовлетворительно**» выставляется в том случае, когда обучающийся правильно выполнил задание лабораторной работы, ответил на теоретические вопросы с существенными неточностями. Показал минимальные удовлетворительные знания и умения при выполнении лабораторной работы в рамках учебного материала.

Оценка «**неудовлетворительно**» выставляется в том случае, когда обучающийся неправильно выполнил задание лабораторной работы, не ответил на теоретические вопросы. Показал недостаточный уровень знаний и умений при выполнении лабораторной работы в рамках учебного материала.

Критерии оценки при проведении зачета

Оценки «**зачтено**» и «**незачтено**» выставляются по дисциплинам, формой заключительного контроля которых является зачет. При этом оценка «зачтено» должна соответствовать параметрам любой из положительных оценок («отлично», «хорошо», «удовлетворительно»), а «незачтено» — параметрам оценки «неудовлетворительно».

Оценка «**отлично**» выставляется обучающемуся, который обладает всесторонними, систематизированными и глубокими знаниями материала учебной программы, умеет свободно выполнять задания, предусмотренные учебной программой, усвоил основную и ознакомился с дополнительной литературой, рекомендованной учебной программой. Как правило, оценка «отлично» выставляется обучающемуся усвоившему взаимосвязь основных положений и понятий дисциплины в их значении для приобретаемой специальности, проявившему творческие способности в понимании, изложении и использовании учебного материала, правильно обосновывающему принятые решения, владеющему разносторонними навыками и приемами выполнения практических работ.

Оценка «хорошо» выставляется обучающемуся, обнаружившему полное знание материала учебной программы, успешно выполняющему предусмотренные учебной программой задания, усвоившему материал основной литературы, рекомендованной учебной программой. Как правило, оценка «хорошо» выставляется обучающемуся, показавшему систематизированный характер знаний по дисциплине, способному к самостоятельному пополнению знаний в ходе дальнейшей учебной и профессиональной деятельности, правильно применяющему теоретические положения при решении практических вопросов и задач, владеющему необходимыми навыками и приемами выполнения практических работ.

Оценка «удовлетворительно» выставляется обучающемуся, который показал знание основного материала учебной программы в объеме, достаточном и необходимым для дальнейшей учебы и предстоящей работы по специальности, справился с выполнением заданий, предусмотренных учебной программой, знаком с основной литературой, рекомендованной учебной программой. Как правило, оценка «удовлетворительно» выставляется обучающемуся, допустившему погрешности в ответах на экзамене или выполнении экзаменационных заданий, но обладающему необходимыми знаниями под руководством преподавателя для устранения этих погрешностей, нарушающему последовательность в изложении учебного материала и испытывающему затруднения при выполнении практических работ.

Оценка «неудовлетворительно» выставляется обучающемуся, не знающему основной части материала учебной программы, допускающему принципиальные ошибки в выполнении предусмотренных учебной программой заданий, неуверенно с большими затруднениями выполняющему практические работы. Как правило, оценка «неудовлетворительно» выставляется обучающемуся, который не может продолжить обучение или приступить к деятельности по специальности по окончании университета без дополнительных занятий по соответствующей дисциплине.

Критерии оценки при проведении экзамена

Оценка «отлично» выставляется обучающемуся, который обладает всесторонними, систематизированными и глубокими знаниями материала учебной программы, умеет свободно выполнять задания, предусмотренные учебной программой, усвоил основную и ознакомился с дополнительной литературой, рекомендованной учебной программой. Как правило, оценка «отлично» выставляется обучающемуся усвоившему взаимосвязь основных положений и понятий дисциплины в их значении для приобретаемой специальности, проявившему творческие способности в понимании, изложении и использовании учебного материала, правильно обосновывающему принятые решения, владеющему разносторонними навыками и приемами выполнения практических работ.

Оценка «хорошо» выставляется обучающемуся, обнаружившему полное знание материала учебной программы, успешно выполняющему предусмотренные учебной программой задания, усвоившему материал основной литературы, рекомендованной учебной программой. Как правило, оценка «хорошо» выставляется обучающемуся, показавшему систематизированный характер знаний по дисциплине, способному к самостоятельному пополнению знаний в ходе дальнейшей учебной и профессиональной деятельности, правильно применяющему теоретические положения при решении практических вопросов и задач, владеющему необходимыми навыками и приемами выполнения практических работ.

Оценка «удовлетворительно» выставляется обучающемуся, который показал знание основного материала учебной программы в объеме, достаточном и необходимым для дальнейшей учебы и предстоящей работы по специальности, справился с выполнением заданий, предусмотренных учебной программой, знаком с основной литературой, рекомендованной учебной программой. Как правило, оценка «удовлетворительно» выставляется обучающемуся, допустившему погрешности в ответах на экзамене или выполнении экзаменационных заданий, но обладающему необходимыми знаниями под руководством преподавателя для устранения этих погрешностей, нарушающему последовательность в изложении учебного материала и испытывающему затруднения при выполнении практических работ.

Оценка «неудовлетворительно» выставляется обучающемуся, не знающему основной части материала учебной программы, допускающему принципиальные ошибки в выполнении предусмотренных учебной программой заданий, неуверенно с большими затруднениями выполняющему практические работы. Как правило, оценка «неудовлетворительно» выставляется обучающемуся, который не может продолжить обучение или приступить к деятельности по специальности по окончании университета без дополнительных занятий по соответствующей дисциплине.

Контроль освоения дисциплины «Программирование» проводится в соответствии с Положением системы менеджмента качества КубГАУ 2.5.1 «Текущий контроль успеваемости и промежуточная аттестация обучающихся».

8 Перечень основной и дополнительной литературы

Основная учебная литература:

1. Программирование на языке Си++: учеб. пособие [Электронный ресурс] / А.Г. Мурлин, В.А. Мурлина, Н.В. Ефанова, Е.А. Иванова. – Краснодар, КубГАУ, 2016. – 186 с. – Режим доступа: https://edu.kubsau.ru/file.php/118/Programmirovanie_uchebnoe_posobie_2016.pdf

2. Ковалевская, Е. В. Методы программирования : учебное пособие / Е. В. Ковалевская, Н. В. Комлева. — Москва : Евразийский открытый институт, 2011. — 320 с. — ISBN 978-5-374-00356-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/10784.html>

3. Лойко В.И. Алгоритмы и структуры данных : учеб. пособие [Электронный ресурс] / В.И. Лойко, Н.В. Ефанова, Е.А. Иванова. — Краснодар : КубГАУ, 2018. — 206 с. — Режим доступа: https://edu.kubsau.ru/file.php/118/AiSD_1.pdf

Дополнительная учебная литература:

1. Смирнов, А. А. Технологии программирования : учебное пособие / А. А. Смирнов, Д. В. Хрипков. — Москва : Евразийский открытый институт, 2011. — 191 с. — ISBN 978-5-374-00296-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/10900.html>

2. Объектно-ориентированное программирование: учеб. пособие [Электронный ресурс] / Е. А. Иванова, Н. В. Ефанова, Т. А. Крамаренко. — Краснодар : КубГАУ, 2018. — 86 с. — Режим доступа: https://edu.kubsau.ru/file.php/118/Uch_posobie_OOP_2018_403992_v1_.PDF

3. Курипта, О. В. Основы программирования и алгоритмизации [Электронный ресурс] : практикум / О. В. Курипта, О. В. Минакова, Д. К. Проскурин. — Электрон. текстовые данные. — Воронеж : Воронежский государственный архитектурно-строительный университет, ЭБС АСВ, 2015. — 133 с. — ISBN 978-5-89040-575-3. — Режим доступа: <http://www.iprbookshop.ru/59123.html>

9 Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень ЭБС

№	Наименование	Тематика	Ссылка
1.	IPRbook	Универсальная	http://www.iprbookshop.ru/
2.	Образовательный портал КубГАУ	Универсальная	https://edu.kubsau.ru/

Рекомендуемые интернет сайты:

— образовательный портал КубГАУ [Электронный ресурс]: Режим доступа: <http://edu.kubsau.local>

— материалы Национального Открытого Университета «Интуит» [Электронный ресурс]: Режим доступа: <http://www.intuit.ru>

- материалы учебного центра «Специалист» при МГТУ им. Н.Э. Баумана [Электронный ресурс]: Режим доступа: <http://www.specialist.ru>;
- материалы сайта образовательной платформы Coursera [Электронный ресурс]: Режим доступа: <http://www.coursera.org>;
- материалы портала «Открытое образование» [Электронный ресурс]: Режим доступа: <http://openedu.ru>;
- материалы портала для разработчиков Microsoft [Электронный ресурс]: Режим доступа: <http://msdn.microsoft.com>.

10 Методические указания для обучающихся по освоению дисциплины

Перечень учебно-методической литературы по освоению дисциплины:

1. Крамаренко Т. А. Программирование : практикум. Ч. 1 [Электронный ресурс] / Т. А. Крамаренко, Е. А. Иванова. – Краснодар : КубГАУ, 2020. – 171 с. – Режим доступа: https://edu.kubsau.ru/file.php/118/Praktikum_Programmirovanie_2020_chast1_593606_v1_.PDF
2. Крамаренко Т. А. Программирование : практикум. Ч. 2 [Электронный ресурс] / Т. А. Крамаренко, Е. А. Иванова. – Краснодар : КубГАУ, 2020. – 87 с. – Режим доступа: https://edu.kubsau.ru/file.php/118/Praktikum_Programmirovanie_2020_chast2_594210_v1_.PDF.
3. Программирование : метод. рекомендации по организации самостоятельной работы [Электронный ресурс] / сост. Т. А. Крамаренко, Е. А. Иванова. – Электрон. текстовые данные. – Краснодар : КубГАУ, 2020. – 28 с. – Режим доступа: https://edu.kubsau.ru/file.php/118/Metodichka_po_SR_Programmirovanie_594203_v1_.PDF.
4. Программирование на СИ/СИ++. Методические указания к лабораторным работам по дисциплине «Программирование»: Часть 1 [Электронный ресурс] / А.Г. Мурлин, Н.В. Ефанова, Е.А. Иванова, О.М. Гончар. – Краснодар: КубГАУ, 2012. – 119 с. - Режим доступа: https://edu.kubsau.ru/file.php/118/01_Programmirovanie_chast_1_.pdf
5. Программирование на СИ/СИ++. Методические указания к лабораторным работам по дисциплине «Программирование»: Часть 2 [Электронный ресурс] / А.Г. Мурлин, Н.В. Ефанова, Е.А. Иванова, О.М. Гончар. – Краснодар: КубГАУ, 2012. – 118 с. - Режим доступа: https://edu.kubsau.ru/file.php/118/02_Programmirovanie_chast_2_.pdf

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине позволяют: обеспечить взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействие посредством сети "Интернет"; фиксировать ход образовательного процесса, результатов промежуточной аттестации по дисциплине и результатов освоения образовательной программы; организовать процесс образования путем визуализации изучаемой информации посредством использования презентационных технологий; контролировать результаты обучения на основе компьютерного тестирования.

11.1 Перечень программного обеспечения

№	Наименование	Краткое описание
1	Windows	Операционная система
3	Office	Пакет офисных приложений
4	INDIGO	Тестирование
3	Microsoft Visual Studio	Разработка приложений
4	Microsoft Visio	Схемы и диаграммы

11.2 Перечень современных профессиональных баз данных, информационных справочных и поисковых систем

№	Наименование	Тематика	Электронный адрес
1.	Научная электронная библиотека «eLIBRARY.RU»	Универсальная	https://elibrary.ru

11.3 Доступ к сети Интернет

Доступ к сети Интернет, доступ в электронную информационно-образовательную среду университета

12 Материально-техническое обеспечение для обучения по дисциплине

Планируемые помещения для проведения всех видов учебной деятельности.

№ п/п	Наименование учебных предметов, курсов, дис- циплин (модулей), прак- тики, иных видов учеб- ной деятельности, предусмотренных учеб- ным планом образова- тельной программы	Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помеще- ния для самостоятельной ра- боты, с указанием перечня основного оборудования, учебно-наглядных пособий и используемого программ- ного обеспечения	Адрес (местоположение) по- мещений для проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается наименование ор- ганизации, с которой заключен договор)
1	2	3	4
1	Программирование	Помещение №112 ЗР, посадочных мест — 96; площадь — 49,7кв.м; учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Специализированная мебель (учебная доска, учебная мебель); технические средства обучения, наборы демонстрационного оборудования и учебно-наглядных пособий (ноутбук, проектор, экран); программное обеспечение: Windows, Office. Помещение №403 ЭК, посадочных мест — 50; площадь — 83,5кв.м; учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Сплит-система — 2 шт.; специализированная мебель (учебная доска, учебная мебель); технические средства обучения, наборы демонстрационного оборудования и учебно-наглядных пособий (ноутбук, проектор, экран); программное обеспечение: Windows, Office. Помещение №215 ЭК, посадочных	350044, Краснодарский край, г. Краснодар, ул. им. Калинина, 13

		мест — 20; площадь — 44 кв.м; учебная аудитория для проведения занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Технические средства обучения (компьютер персональный — 10 шт.); доступ к сети «Интернет»; доступ в электронную информационно-образовательную среду университета; специализированная мебель (учебная доска, учебная мебель). программное обеспечение: Windows, Office. Microsoft Visual Studio, Microsoft Visio Помещение №216 ЭК, посадочных мест — 30; площадь — 41,8 кв.м; учебная аудитория для проведения занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Технические средства обучения (компьютер персональный — 10 шт.); доступ к сети «Интернет»; доступ в электронную информационно-образовательную среду университета; специализированная мебель (учебная доска, учебная мебель). программное обеспечение: Windows, Office. Microsoft Visual Studio, Microsoft Visio Помещение №4 ЭК, площадь — 31,1 кв.м; помещение для хранения и профилактического обслуживания учебного оборудования. кондиционер — 2 шт.; лабораторное оборудование	
--	--	--	--

		(шкаф лабораторный — 1 шт.; набор лабораторный — 1 шт.); технические средства обучения (принтер — 1 шт.; проектор — 1 шт.; микрофон — 1 шт.; ибп — 4 шт.; сервер — 1 шт.; носитель информации — 1 шт.; компьютер персональный — 15 шт.). Помещение №4 ЭК, площадь — 9,1 кв.м; помещение для хранения и профилактического обслуживания учебного оборудования. сплит-система — 2 шт.; штатив — 1 шт.; лабораторное оборудование (шкаф лабораторный — 2 шт.; стенд лабораторный — 4 шт.); технические средства обучения (экран — 1 шт.; сетевое оборудование — 5 шт.; сервер — 6 шт.; компьютер персональный — 2 шт.).	
2	Программирование	Помещение №206 ЭК, посадочных мест — 20; площадь — 41 кв.м; помещение для самостоятельной работы. Технические средства обучения (компьютер персональный — 9 шт.); доступ к сети «Интернет»; доступ в электронную информационно-образовательную среду университета; специализированная мебель (учебная мебель). Программное обеспечение: Windows, Office, специализированное лицензионное и свободно распространяемое программное обеспечение, предусмотренное в рабочей программе	350044, Краснодарский край, г. Краснодар, ул. им. Калинина, 13