

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Петровская Анна Викторовна

Должность: Директор

Дата подписания: 16.01.2026 11:17:32

Уникальный программный ключ:

798bda6555fbdebe827768f6f1710bd17a9070c91a10baa5a5110e86319
к основной профессиональной образовательной программе
по направлению подготовки 09.03.03 Прикладная информатика
направленность (профиль) программы Прикладная информатика в
экономике

Приложение 6

**Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский экономический университет имени Г.В. Плеханова»**

Краснодарский филиал РЭУ им. Г. В. Плеханова

Факультет экономики, менеджмента и торговли

Кафедра экономики и цифровых технологий

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине Алгоритмизация и программирование

Направление подготовки

09.03.03 Прикладная информатика

Направленность (профиль) программы

Прикладная информатика в экономике

Уровень высшего образования

Бакалавриат

Год начала подготовки 2025

Краснодар – 2024 г.

Составитель:

к.к., доцент кафедры экономики и цифровых технологий Л.В. Кухаренко

Оценочные материалы одобрены на заседании кафедры экономики и цифровых технологий Краснодарского филиала РЭУ им. Г.В. Плеханова протокол № 9 от 14 марта 2024 г.

Оценочные материалы составлены на основе оценочных материалов по дисциплине «Алгоритмизация и программирование», утвержденной на заседании базовой кафедры цифровой экономики института развития информационного общества федерального государственного бюджетного образовательного учреждения высшего образования «Российский экономический университет имени Г.В. Плеханова» протокол № 11 от 17 мая 2021 г., разработанной авторами:

Ивановым Е.А., ст. преподаватель, базовой кафедры цифровой экономики института развития информационного общества

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине «Алгоритмизация и программирование»

ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ С УКАЗАНИЕМ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ И ЭТАПОВ ИХ ФОРМИРОВАНИЯ ПО ДИСЦИПЛИНЕ

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций (код и наименование индикатора)	Результаты обучения (знания, умения)	Наименование контролируемых разделов и тем
ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Разрабатывает и реализует эффективные алгоритмы и программы с использованием современных технологий и языков программирования	ОПК-7.1. 3-1. Знает теоретические основы алгоритмизации и программирования	Тема 1. Введение в алгоритмизацию и программирование Тема 2. Введение в программирование на языках Си/Си++ Тема 3. Углубленное рассмотрение языка программирования
		ОПК-7.1. 3-2. Знает основные языки программирования и современные программные среды разработки информационных систем и технологий	Тема 3. Углубленное рассмотрение языка программирования Тема 4. Функции стандартных библиотек Тема 5. Особенности работы со строками и файлами Тема 6. Особенности работы со структурированной информацией
		ОПК-7.1. 3-3. Знает современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования	Тема 3. Углубленное рассмотрение языка программирования Тема 4. Функции стандартных библиотек Тема 5. Особенности работы со строками и файлами Тема 6. Особенности работы со структурированной информацией
		ОПК-7.1. 3-4. Знает технологию разработки и отладки программ	Тема 3. Углубленное рассмотрение языка программирования Тема 4. Функции стандартных библиотек Тема 5. Особенности работы со строками и файлами Тема 6. Особенности работы со структурированной информацией
		ОПК-7.1. 3-5. Знает синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных	Тема 2. Введение в программирование на языках Си/Си++ Тема 3. Углубленное рассмотрение языка программирования Тема 4. Функции стандартных библиотек Тема 5. Особенности работы со строками и файлами Тема 6. Особенности работы со структурированной информацией
		ОПК-7.1. У-1. Умеет выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости	Тема 1. Введение в алгоритмизацию и программирование Тема 2. Введение в программирование на языках Си/Си++ Тема 3. Углубленное рассмотрение языка программирования Тема 4. Функции стандартных библиотек

		от типа прикладной задачи	Тема 5. Особенности работы со строками и файлами Тема 6. Особенности работы со структурированной информацией
		ОПК-7.1. У-2. Умеет программировать приложения и создавать программные прототипы решения прикладных задач	Тема 2. Введение в программирование на языках Си/Си++ Тема 3. Углубленное рассмотрение языка программирования Тема 4. Функции стандартных библиотек Тема 5. Особенности работы со строками и файлами Тема 6. Особенности работы со структурированной информацией
		ОПК-7.1. У-3. Умеет разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика	Тема 2. Введение в программирование на языках Си/Си++ Тема 3. Углубленное рассмотрение языка программирования Тема 4. Функции стандартных библиотек Тема 5. Особенности работы со строками и файлами Тема 6. Особенности работы со структурированной информацией

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ХАРАКТЕРИЗУЮЩИЕ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

Перечень учебных заданий на аудиторных занятиях

Вопросы для проведения опроса во время аудиторных занятий

Тема 1. Введение в алгоритмизацию и программирование

- Инструментальные средства разработки программ.
- Сегментация приложения Си++.
- Базовые типы данных, зависимость от разрядности компилятора.
- Виды операций и их приоритет.
- Операторы цикла.
- Операторы условных и безусловных переходов.

Тема 2. Введение в программирование на языках Си/Си++

- Развитие языков программирования.
- Шаблоны приложений Microsoft Visual Studio.
- Структура программы на языке Си.
- Размер типа в зависимости от разрядности компилятора.
- Составной оператор.
- Нетрадиционные операции языка Си++.
- Особенности использования тернарного оператора.

Тема 4. Функции стандартных библиотек

- Генератор псевдослучайных чисел.
- Преобразование числовых значений в строковые и наоборот.
- Достаточно ли математических функций стандартной библиотеки для решения любых математических задач?
- Функции работы с памятью.
- Функции поддержки арифметики с плавающей точкой.

Тема 5. Особенности работы со строками и файлами

- Стока как массив символов.
- Виды объявления строк.
- Безопасные функции.
- Почему при объявлении строки в виде указателя может возникнуть ошибка «access violation»?
- В каких случаях используются текстовые, а в каких двоичные потоки?
- Для чего нужен указатель в файле и в каких случаях требуется его перемещение?
- Для чего используются стандартные файлы?

Вопросы для проведения групповой дискуссии во время аудиторных занятий

Тема 3. Углубленное рассмотрение языка программирования.

- Особенности организации многомерных массивов.
- Связь массивов и указателей.

- Отличие функции от составного оператора.
- Применение рекурсии.
- Функции с переменным числом аргументов.

Тема 6. Особенности работы со структурированной информацией

- Классификация потоков по направлению обмена.
- Классификация потоков по виду устройств.
- Стандартные манипуляторы.
- Методы неформатированного чтения и записи в поток.
- Отличия использования функций стандартной библиотеки работы с файлами и файловых потоков, достоинства и недостатки.

Задания для текущего контроля

Комплект тестов/тестовых заданий по темам 1-3

Индикаторы достижения: ОПК-7.1

Тестовые задания закрытого типа

1. Тестовое задание закрытого типа с выбором одного правильного ответа

1.1. Принцип фон Неймана предполагает хранение программы ...

- На внешнем носителе
- В памяти компьютера
- На специальной плате в виде перемычек
- В кэш-памяти процессора

1.2. Выполнение программы на языке C++ начинается с:

- первой строки;
- первой функции;
- функции main;
- подключения заголовочного файла;

1.3. Инструкция ... вставляет в программу заранее подготовленные тексты из включаемых файлов:

- #include;
- #define;
- #ifndef;
- #else;

1.4. Тип результата при сложении переменных типа short:

- short;
- int;
- long;
- float;

1.5. Фрагмент программы: int S = 0; for (int i = 0; i < N; i++) S += a[i]; S /= N;

- находит сумму элементов вектора;
- находит среднее арифметическое элементов вектора;
- уменьшает элементы вектора вдвое;
- заполняет вектор нулевыми значениями;

1.6. Оператор `continue` используется для:

- а) продолжения выполнения программы после её остановки отладчиком;
- б) преждевременного завершения текущей итерации цикла и перехода к новой;
- в) выхода из цикла;
- г) выхода из программы;

1.7. Функция `main` - это:

- а) функция определения основания логарифма;
- б) главная функция программы;
- в) основная функция для отладки программы;
- г) основная функция для каждого файла, включенного в программу;

1.8. При правильном выполнении программы в операционную систему передается:

- а) нулевой результат;
- б) ненулевой результат;
- в) отрицательный результат;
- г) никакой результат не передается;

1.9. При записи значения для элемента 25 в массиве из 24-х членов:

- а) программа не будет компилироваться;
- б) значение будет добавлено в последнюю ячейку памяти, принадлежащую массиву;
- в) значение не будет никуда добавлено;
- г) значение будет добавлено в ячейку памяти, не принадлежащую массиву;

1.10. Отсутствие возвращаемого значения функции обозначается как:

- а) список параметров функции через запятую, заключенный в круглые скобки;
- б) возвращаемое значение;
- в) ключевое слово `void`;
- г) ключевое слово `return`;

2. Тестовое задание закрытого типа с выбором нескольких правильных ответов

2.1. Какие основные устройства содержит ЭВМ неймановской структуры?

- а) арифметико-логическое устройство;
- б) устройство управления;
- в) устройства ввода-вывода;
- г) запоминающее устройство;
- д) устройство контроля.

2.2. Отметьте способы описания алгоритмов.

- а) блок-схема;
- в) с помощью сетей Петри;
- б) словесно-формульный;
- г) помощью нормальных форм.

2.3. Отметьте те понятия, которые связаны с понятием «информатика».

- а) сигнал;
- б) вещество;
- в) сообщение;
- г) данные;
- д) энергия.

2.4. Какие из выражений имеют значение true?

- а) $(1 > 3) \parallel !(1 > 3)$
- б) $(1 < 3) \&\& (2 == 1)$
- в) $(1 < 3) \parallel (2 == 1)$
- г) $!((1 < 3) \&\& (2 == 1))$
- д) $(!(1 > 3)) \&\& (1 < 3)$

2.5 К лексемам языка относятся

- а) идентификаторы;
- в) ключевые слова;
- б) числа;
- г) константы.
- д) символы

2.6 Укажите вещественные типы языка C++

- а) int;
- б) double;
- в) float;
- г) void
- д) char

2.7 Какие из выражений дадут одинаковый результат для любых значений переменной A

- а) $1 + + A;$
- б) $A + 1;$
- в) $A + -;$
- г) $A + - = 1$
- д) $A + +$

2.8. Найдите операторы вывода, не содержащие ошибок для вывода значений переменных (int k; float x,y)

- а) printf ("*%4f %d %4.2f*", x, k, y);
- б) printf ("*%f %d %4f*", x, k, y);
- в) printf ("*%4f %4d %4.2f*", x, k, y).

2.9. Выберите все правильные утверждения.

- а) результат операций сравнения действительных чисел — действительный
- б) результат арифметических операций над действительными числами — действительный
- в) в одном выражении можно использовать операнды действительного и целого типов

2.10. Выберите все правильные ответы. Элементарные конструкции (лексемы) языка C++ включают в себя:

- а) знаки операций
- б) ключевые слова
- в) двоичные числа
- г) строковые константы

Тестовые задания открытого типа

3. Тестовые задания открытого типа, т.е. с указанием ответов

3.1 Общество, в котором большинство работающих занято производством, хранением, переработкой и реализацией информации, особенно ее высшей формы — знаний называется
(Ответ записать словом в именительном падеже)

3.2 ... – организованный социально-экономический и научно-технический процесс создания оптимальных условий для удовлетворения информационных потребностей и реализации прав граждан, органов государственной власти, органов местного самоуправления, организаций на основе формирования и использования информационных ресурсов с помощью средств вычислительной техники

(Ответ записать словом в именительном падеже)

3.3 Система точно сформулированных правил, определяющая процесс преобразования допустимых исходных данных в желаемый результат, называется **алгоритм**

3.4. ... – это отдельные факты, характеризующие объекты, процессы, явления, признаки или записанные наблюдения, которые по каким-то причинам не используются, а только хранятся.

3.5. Двумя байтами можно представить ... разных значений.

(Ответ записать целым числом)

3.6. В качестве имени объекта (переменной, константы, функции и т.д.) используется ...

3.7. Объект, который в ходе выполнения программы может менять свое значение называется ...

3.8. После выполнения фрагмента программы

```
*y = 7;  
*x = 5;  
*x *= 5;  
(*x)++;  
z = x + y;
```

значение переменной z будет равно:

3.9. Запишите лексему, которой обозначается операция присваивания в Си++

3.10. Какими станут значения переменных I и K после выполнения фрагмента программы
 $K=3; I=3; K=++i$ (сначала укажите значение I, затем через пробел значение K)

3.11. Какими станут значения переменных I и K после выполнения фрагмента программы
 $K=3; I=3; K=i++$ (сначала укажите значение I, затем через пробел значение K)

3.12 Чему будет равен результат вычисления выражений фрагмента программы:

```
int d=5;  
bool b = true, c;  
c = (!b||(d>3));
```

3.13 Что будет выведено в результате выполнения данного кода?

```
int f1(int x1, int &x2)  
{  
    return ++x1 + (++x2);  
}  
int main()  
{  
    int a = 7, k = 1;
```

```
k = f1(a, k);
cout << a << " " " " << k;
}
```

3.14 С помощью какого метода можно изменить текущую позицию в файле?

seekp

3.15 Какой результат будет выведен после выполнения фрагмента программы?

```
int m = 1, n=2;
double A = (double) m/n;
std::cout << A;
```

3.16 Определите размер структуры в байтах

```
struct {char fio[30];
unsigned char date:4;
unsigned char code:4;
};
```

3.17 Что будет выведено в результате выполнения следующего кода?

```
int a[4] = { 1,2,3,4};
int* p = a;
cout << (*p+2) + *p;
```

3.18 Какой результат будет выведен после выполнения фрагмента программы?

```
int *a;
int b[2];
a = b;
b[0] = 7;
b[1] = 10;
*a++;
cout << *a;
```

3.19 Задан массив: int m[3][4] = { { 1, 2, 3, 4}, { 5, 6, 7, 8}, { 9, 10, 11, 12} }; Какое значение содержится в элементе m[2][3]?

3.20 Если имеется код int x; cin >> x; и вводится «1.2», то что будет записано в переменной x?

3.21 Какой результат будет после выполнения фрагмента программы?

```
int main()
{
char *s1;
char s2[] = "Hello";
*s1 = "Hi!";
s1[2] = s2[1];
cout << s1;
}
```

3.22 Что будет выведено на экран в результате работы следующего фрагмента программы:

```
double x = 123.456789;
printf("%g\n",x);
```

3.23 Чему равно значение выражения (a && ! b || c), где a, b и c - величины типа bool, имеющие значения false, true и true соответственно?

3.24 Чему равно значение выражения ($a \parallel b \&\& a \parallel c$), где a , b и c — величины типа `bool`, имеющие значения `false`, `true` и `true` соответственно?

3.25 Что будет выведено на экран в результате работы программы (учитывая формат вывода)?

```
#include main()
```

```
{  
int i;  
float t;  
i = 1;  
t = 5;  
printf("%3.1f%d%2d", t, i, i);  
}
```

3.26 Что будет выведено на экран в результате работы программы?

```
#include <stdio.h>
```

```
int main()  
{  
const char a[] = "abcrcaab";  
int i;  
for (i = 0; i < 8; i++)  
if (a[i] < 'c')  
printf("%c", a[i]);  
}
```

3.27 В массиве A 132 элемента, напишите правильное обращение к последнему элементу массива?

3.28. Что будет выведено на экран после выполнения данного кода

```
{  
int ar[4];  
ar[0]=1;  
ar[1]=2;  
ar[2]=3;  
ar[3]=4;  
for (int i = 0; i<=3; i++)  
cout << ar[i];  
}
```

3.29. Двумерный массив Ar , представленный в виде матрицы, имеет вид

$$\begin{bmatrix} 1 & 2 & 2 \\ 5 & 6 & 7 \\ 9 & 1 & 2 \end{bmatrix}$$

Что будет выведено на экране в результате выполнения кода

```
int flag=7;  
for (int i = 0; i<3; i++) {  
if (flag)  
    cout << Ar[1][i];  
else  
    cout << "Error";
```

}

3.30 Что будет выведено на экран в результате следующего выражения?

```
struct my
{
    int a, b;
} m1;
int func(my f)
{
    return f.a + f.b++;
}
int main()
{
    m1.a = 5; m1.b = 10;
    cout << func(m1);
    return 0;
}
```

Ответы (ключи) на тестовые задания представить в форме таблицы

Номер во- проса	Варианты ответа	Номер во- проса	Варианты ответа	Номер во- проса	Варианты от- вета
1.1	б	1.2	в	1.3	а
1.4	б	1.5	б	1.6	б
1.7	б	1.8	а	1.9	в
1.10	в				
2.1	а,б,в,г	2.2	А,б	2.3	А,в,г
2.4	А,в,г,д	2.5	А,в,г	2.6	Б,в
2.7	Б,г,д	2.8	А,б,в	2.9	Б,в
2.10	А,б,г				
3.1	информационное	3.2	Информатизация общества	3.3	алгоритм
3.4	данные	3.5	65536	3.6	идентификатор
3.7	переменная	3.8	33	3.9	==
3.10	4 4	3.11	4 3	3.12	true
3.13	7 10	3.14	seekp	3.15	0.5
3.16	31	3.17	4	3.18	10
3.19	12	3.20	1	3.21	Hie
3.22	123.457	3.23	true	3.24	true
3.25	5.01 1	3.26	abaab	3.27	A[131]
3.28	1234	3.29	567	3.30	15

Критерии оценки (в баллах):

- 8 баллов выставляется обучающемуся, если даны правильные ответы на 45 и более вопросов;
- 7 баллов выставляется обучающемуся, если даны правильные ответы на 40 и более вопросов;
- 6 баллов выставляется обучающемуся, если даны правильные ответы на 35 и более вопросов;
- 5 баллов выставляется обучающемуся, если даны правильные ответы на 30 и более вопросов;
- 4 балла выставляется обучающемуся, если даны правильные ответы на 25 и более вопросов;
- 3 балла выставляется обучающемуся, если даны правильные ответы на 20 и более вопросов;
- 2 балла выставляется обучающемуся, если даны правильные ответы на 15 и более вопросов;

- 1 балл выставляется обучающемуся, если даны правильные ответы на 7 и более вопросов;
- 0 баллов выставляется обучающемуся, если не даны правильные ответы на 7 вопросов.

Комплект заданий для контрольной работы по темам 4-6

Индикаторы достижения: ОПК-7.1

Задание 1

Тема 4. Функции стандартных библиотек

Написать игру "Угадай число"

Компьютер используя функцию rand() загадывает случайное число от 1 до 100

Игрок вводит числа, пока не угадает число компьютера.

При вводе числа, которое меньше загаданного выводится сообщение "Ваше число меньше", если число больше, то "Ваше число больше".

Ввод организуется с помощью бесконечного цикла, с выходом по оператору break в случае угадывания.

Задание 2

Тема 5. Особенности работы со строками и файлами

Ввести строку символов (предусмотреть ввод пустой строки или строки из одних пробелов). Разделителями слов в строке являются пробелы (один или несколько).

1. Определить количество слов строки, начинающихся и заканчивающихся на одну и ту же букву.
2. Удалить слова с заданным количеством символов.
3. Составить массив из слов, в которых ни одна буква не повторяется.
4. Составить массив из номеров слов, в которых есть повторяющиеся символы.
5. Поменять местами i и j слово (номера слов i и j задаются при вводе).
6. Поменять местами слово максимальной и слово минимальной длины.
7. Определить слово максимальной длины, в котором все символы различны.
8. Определить слова, в которых содержится больше двух гласных букв русского алфавита.
9. Определить номера слов, в которых содержится больше одного цифрового символа.
10. Определить номер слова-палиндрома, имеющего максимальную длину.
11. Определить слова, в которых все буквы повторяются.
12. Определить номера слов, состоящих из тех же символов, что и последнее слово строки.
13. Составить массив из слов, в которых все имеющиеся в них цифровые символы образуют монотонную последовательность.
14. Удалить все слова-палиндромы, имеющие максимальную длину.
15. Продублировать через один пробел слова-палиндромы, в которых нет ни одного цифрового символа.

Задание 3

Тема 6. Особенности работы со структурированной информацией

1. Комплексные числа состоят из действительной и мнимой частей, обе являются вещественными числами. Например, **1.5+0.3i**. Здесь **1.5** - действительная часть, **0.3** - мнимая часть, **i** - так называемая мнимая единица. Части комплексных чисел могут быть отрицательными величинами. Например, **-3+1.5i**, **2-2i**, **-10.5-1.25i**.

Если даны комплексные числа **a** и **b**, где их действительные части обозначены как **a.x** и **b.x**, а мнимые - как **a.y** и **b.y**, то их сумма и произведение находятся по формулам:

$$a + b = (a.x + b.x) + (a.y + b.y)i$$

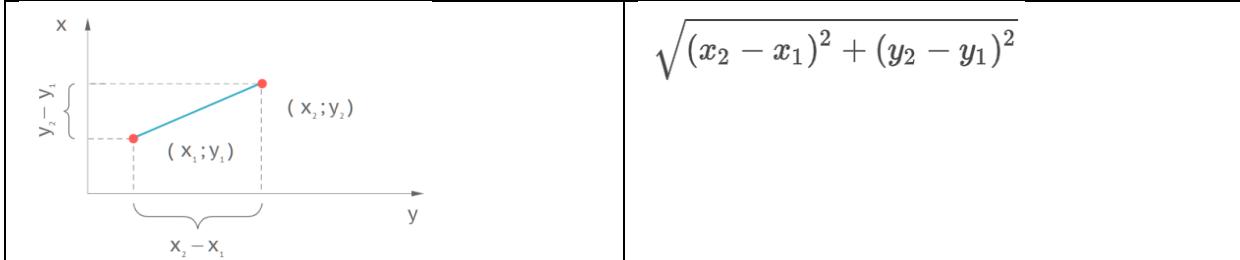
$$a * b = (a.x * b.x - a.y * b.y) + (a.x * b.y + a.y * b.x)i$$

Написать программу сложения и умножения двух комплексных чисел используя структуры.

2. Задана следующая структура и переменные A и B типа этой структуры

```
struct point {  
    float x,y;  
} A, B;
```

Написать программу, в которой задаются координаты точек в переменные A и B, а затем вычисляется расстояние между точками.



3. Создать массив структур, каждая из которых содержит заданную информацию о товаре в компьютерном магазине. Пример

```
struct item {  
    char name[50];  
    unsigned short qty;  
    unsigned int price;  
};
```

В начале программы массив инициализируется следующим образом:

```
struct item items[5] = {  
    "Pentium G3220", 8, 4100,  
    "AMD FX-6300", 6, 5700,  
    "Core-i3-4330", 9, 6500,  
    "Core i5-3450", 5, 7400  
    "Core i5-4670K", 3, 9500,  
};
```

При запуске программы должен выводиться список товаров с данными об их количестве и цене. Далее пользователю должно быть предложено скорректировать информацию о количестве и цене. После изменения данных должен выводиться обновленный список товаров.

Критерии оценки (в баллах):

- 12 баллов выставляется обучающемуся, если все задания выполнены корректно в соответствии с предъявляемыми требованиями; обучающийся **знает верно и в полном объеме** теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет верно и в полном объеме** выбирать и применять совре-

менные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программирует приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

- 7–11 баллов выставляется обучающемуся, если задания выполнены корректно в соответствии с предъявляемыми требованиями, но есть незначительные недоработки; обучающийся **знает с незначительными замечаниями** теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет с незначительными замечаниями** выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программирует приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

- 1–6 баллов выставляется обучающемуся, если одно из заданий не выполнено; обучающийся **знает на базовом уровне, с ошибками** теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет на базовом уровне, с ошибками** выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программирует приложения и создавать программные прототипы решения прикладных задач; разрабатывает программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

- 0 баллов выставляется обучающемуся, если более одного задания не выполнено или имеются существенные недоработки в двух и более заданиях; обучающийся **знает верно и в полном объеме** теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет верно и в полном объеме** выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программирует приложения и создавать программные прототипы решения прикладных задач; разрабатывает программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

Задания для творческого рейтинга

Темы индивидуальных проектов

Индикаторы достижения: ОПК-7.1

Задание 1. Тема 3. Углубленное рассмотрение языка программирования

Вводится длина одномерного массива и значения его элементов.

1. Составить и вывести на экран новый массив с номерами элементов исходного массива, которые равны заданному значению. Заданное значение вводится с клавиатуры.
2. Поменять местами максимальный и минимальный элементы массива. Вывести измененный массив на экран.
3. Все элементы массива, меньшие заданного значения, и их номера записать в новые массивы. Вывести новые массивы на экран. Заданное значение вводится с клавиатуры.
4. Вводится дополнительный массив разрешенных значений. Определить и вывести на экран, сколько элементов исходного массива имеют разрешенные значения.
5. Вводится дополнительный массив разрешенных значений. Составить массив из элементов исходного массива, имеющих неразрешенные значения. Вывести результативный массив на экран.

Ввести размер прямоугольной матрицы и значения ее элементов

1. Построить вектор B, где B(i) –число неотрицательных элементов в i-ой строке.
2. Построить вектор C, где C(j) –среднее арифметическое положительных элементов в j-ом столбце.
3. Построить вектор B, где B(i) –минимальное значение в i-ой строке.
4. Построить вектор C, где C(j) –максимальное значение в j-ом столбце.
5. Построить вектор B, где B(i) –номер максимального значения в i-ой строке.

Задание 2. Тема 5. Особенности работы со строками и файлами

1. Напишите программу перевода введенной строки в требуемый регистр.

В программе должна вводиться строка, затем выбираться требуемый регистр, ввод 0 означает нижний регистр, ввод 1 означает верхний регистр.

Используйте функции: **toupper**, **tolower**, **strlen**.

Организуется цикл

`for(int i=0; i<strlen(str); i++)`

Для ввода строки используйте функцию `scanf_s` или поток ввода `cin`.

2. Напишите программу, переворачивающую введенную фразу.

Например, введено

жили у бабуси три веселых гуся

Программа должна вывести

ясугхылесевиртисубаб у илиж

Подсказка: используйте функцию `strlen` и работайте со строкой, как с массивом в цикле от 0 до половины размера строки. Заменяйте символы зеркально относительно центра строки, используйте дополнительную переменную типа `char` для временного хранения символа при замене.

3. Напишите программу, переворачивающую слова во введенной фразе.

Например, введено

жили у бабуси три веселых гуся

Программа должна вывести

илиж у исубабиртхылесевясуг

Подсказка: используйте функцию `strtok_s` для выделения слов из строки, затем переворачивайте слова и вставляйте на место или используйте индексы выделенного слова в цикле замены.

Задание 3. Тема 6. Особенности работы со структурированной информацией

Сложение длинных чисел

Используя динамический массив структур сложить два числа, например:

$x = 134557952499317879$

$y = 79349864365110$

Числа разбить на группы по девять цифр, которые будут являться элементами структур массива.

Числа вводятся как строка, затем разбиваются с конца по 9 символов, эта часть строки преобразуется в число, например, функцией strtod или atoi. Полученное число помещается в структуру массива.

Результат сложения вывести на экран.

Предусмотреть возможность сложения чисел до 200 символов.

Обрабатывать ошибки ввода чисел.

Дополнительно можно сделать разность, умножение и деление.

Пример сложения

134557952 499317879 (число 134557952499317879)
79349 864365110 (число 79349864365110)

Результат сложения двух чисел

134637302(1)363682989 (число 134637302363682989)

Критерии оценки (в баллах):

- 18–20 баллов выставляется студенту, если разработанные программы соответствуют всем требованиям задания, запускаются и выдают правильный результат; обучающийся **знает верно и в полном объеме** теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет верно и в полном объеме** выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программировать приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

- 10–18 баллов выставляется студенту, если программы запускаются, выдают правильный результат, но разработаны с нарушением требований к реализации; обучающийся **знает с незначительными замечаниями** теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет с незначительными замечаниями** выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программировать приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

- 1–9 баллов выставляется студенту, если некоторые программы выдают неправильный результат, но обучающийся может объяснить, какие ошибки в программах могли к этому привести; обучающийся **знает на базовом уровне, с ошибками** теоретические основы алгоритми-

зации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет на базовом уровне, с ошибками** выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программирует приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

- 0 баллов выставляется студенту, если все или некоторые программы выдают неправильный результат и обучающийся не может объяснить, какие ошибки в программах могли к этому привести; обучающийся **знает верно и в полном объеме** теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных; **умеет верно и в полном объеме** выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программирует приложения и создавать программные прототипы решения прикладных задач; разрабатывает программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ХАРАКТЕРИЗУЮЩИЕ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ ВО ВРЕМЯ ПРОВЕДЕНИЯ ПРО- МЕЖУТОЧНОЙ АТТЕСТАЦИИ

Структура экзаменационного билета

<i>Наименование оценочного средства</i>	<i>Максимальное количество баллов</i>
<i>Вопрос 1</i>	<i>13</i>
<i>Вопрос 2</i>	<i>13</i>
<i>Практическое задание</i>	<i>14</i>

Задания, включаемые в экзаменационный билет

Перечень вопросов к экзамену:

1. Базовые понятия информатики – информация, данные, знания, количество информации, объем данных.
2. Потребительские показатели качества информации.
3. Архитектура современных ЭВМ и принципы фон Неймана.
4. Информационные системы, структура и классификация информационных систем.
5. Представление чисел в памяти компьютера.
6. Системы счисления, позиционные и непозиционные, перевод чисел.
7. Алгебра логики, логические операции.
8. Законы логики.
9. Основные понятия и классификация программного обеспечения.
10. Методология подготовки и решения задач на ЭВМ.
11. Алгоритмы и их свойства.
12. Способы описания алгоритмов. Примеры.
13. Понятие языка программирования.
14. Классификация языков программирования.
15. Свойства и функции языков программирования.
16. Основные требования к языкам программирования.
17. Программы-трансляторы. Назначение и виды.
18. Элементы описания языков программирования.
19. Основные алгоритмические конструкции. Примеры.
20. Понятие системы программирования. Их классификация.
21. Инструментальные средства разработки программ.
22. Этапы создания программы на C++.
23. Алфавит и словарь языка C++..
24. Идентификаторы, типы идентификаторов.
25. Правила именования идентификаторов.
26. Структура и компоненты простой программы на языке C++.
27. Типы данных в языке C++.
28. Константы в языке C++. Объявление констант. Примеры.
29. Переменные в языке C++.
30. Объявление и инициализация переменных. Примеры.

31. Операции, порядок и приоритет операций, типы операций в зависимости от количества операндов.
32. Преобразования типов в языке Си/С++ (явные и неявные).
33. Выражения, правила построения.
34. Понятие линейного алгоритма в С++. Примеры.
35. Операторы условия. Примеры.
36. Операторы цикла. Примеры
37. Программирование итерационных циклов.
38. Операторы переходов.
39. Указатели.
40. Массивы (одномерные, многомерные), определение и инициализация. Примеры.
41. Доступ к массиву через указатель.
42. Строки как массивы символов.
43. Функции работы со строками встроенной библиотеки языка. Примеры программ обработки строк.
44. Функции. Формат функции, прототип функции, функции с аргументами.
45. Функции. Передача аргументов функции. Передача массива в виде аргумента.
46. Рекурсивные функции.
47. Функции с переменным числом аргументов.
48. Ввод-вывод в языке Си++.
49. Библиотека функций С++. Их назначение.
50. Библиотека функций общего назначения.
51. Библиотека математических функций.
52. Функции работы с файлами.
53. Функции работы с памятью.
54. Динамическое выделение памяти под переменные и массивы.
55. Функции обработки строк.
56. Функции преобразования данных.
57. Функции преобразования и тестирования символов.
58. Препроцессор. Директивы препроцессора. Макроопределения и макрофункции.
59. Структуры в языках С++. Объявление, элементы, переменные структурного типа.
60. Доступ к элементам структуры. Операции над структурами.
61. Указатели на структуры. Обращение к элементу структуры по указателю.
62. Массивы структур. Примеры.
63. Определения понятия файл. Типы файлов.
64. Стандартные файлы. Режимы открытия файлов.
65. Функции открытия и закрытия файлов. Функции работы с данными файла. Функция перемещения указателя в файле.
66. Определение понятия поток. Классификация потоков.
67. Стандартные потоки. Стандартные манипуляторы. Методы неформатированного чтения и записи.
68. Файловые потоки. Строковые потоки.
69. Обработка ошибок состояния потоков.
70. Объединения. Обращение к элементу объединения.
71. Объединение как элемент структуры.
72. Битовые поля. Обращение к битовым полям.

Практические задания для экзамена:

1. Дано целое число $N (> 1)$, а также первый член A и разность D арифметической прогрессии. Сформировать и вывести массив размера N , содержащий N первых членов данной прогрессии: $A, A+D, A + 2D, A + 3D, \dots$
2. Составить программу, определяющую сумму ряда $S = 1/2 + 2/3 + 3/4 + \dots + (N-1)/N$; Вывести номера членов ряда со значением из диапазона 0.95 ..0.97.
3. Дано целое число. Найти сумму и произведение его цифр.
4. Даны координаты вершин четырехугольника $ABCB$. Найти сумму длин его диагоналей. Данные для ввода: $A(0,1), B(2,5), C(4,8), D(2,0)$. Для вычисления длины диагонали используйте функцию.
5. Сравните соответствующие строки двух файлов. Если строки совпали, то запишите эту строку в результирующий файл. Вывести на экран данные файлы и полученный. Если полученный файл пуст, вывести соответствующее сообщение
6. Дано натуральное число. Определите, сколько четных цифр используется в записи этого числа. Для определения цифр, входящих в число, используйте функцию.
7. Создайте файл символов (символы вводятся с клавиатуры). Выведите на экран символы с k -го по s -ый.
8. Создайте файл, элементами которого являются целые числа. Найдите среднее арифметическое элементов файла.
9. На печать выводится сказка «О попе и его собаке» определенное число раз. (У попа была собака, он ее любил. Она съела кусок мяса – он ее убил. В землю закопал, надпись написал ...). Используйте рекурсивную функцию для вывода сказки N число раз.

$$x_i = \frac{2x_{i-1}}{3} + \frac{a}{3x_{i-1}^2}, x_1 = 1.$$

10. Составьте алгоритм для вычисления $\sqrt[3]{a}$, используя рекурсию
11. Составить программу, собирающую данные об авиакомпаниях и выдающую справку туриstu до запрашиваемого места. Справка должна содержать:
 - название авиакомпании;
 - название рейса;
 - номер рейса;
 - даты вылета (содержатся в массиве);
 - наличие мест в 1 и 2 классах.
12. Для заданного N составить алгоритм вычисления значения выражения:
$$\frac{(1*1)}{(1+(3*3))} * \frac{(2*2)}{(2+(3*3))} * \dots * \frac{(N*N)}{(N+(3*3))}.$$
Для вычисления очередного множителя используйте функцию.
13. Багаж пассажира характеризуется количеством вещей и общим весом вещей. Дан массив, содержащий сведения о багаже нескольких пассажиров. Сведения о багаже каждого пассажира представляют собой запись с двумя полями: одно поле целого типа (количество вещей) и другое - действительного типа (вес в килограммах). Найти багаж, вес вещей в котором отличается не более, чем на 0.3 кг от общего среднего веса вещей пассажиров.
14. В массиве хранятся данные об учениках класса: фамилия, имя, отчество, адрес (улица, дом, квартира) и домашний телефон (если есть). Вывести на экран список учеников до которых нельзя дозвониться.
15. Дан символ C , изображающий цифру или букву (латинскую или русскую). Если C изображает цифру, то вывести строку «digit», если латинскую букву — вывести строку «lat», если русскую — вывести строку «rus»
16. Данна строка длины n , состоящая из латинских букв, цифр и символов знаков. Преобразовать строку, заменив все «:», встречающиеся среди первых $n/2$ символов, звездочками и заменив точками все «!», встречающиеся среди символов, стоящих после $n/2$ символов

17. Задана строка произвольной длины. Подсчитать в строке число букв А и В, если буква А больше, чем В, то удалить в строке все символы В, иначе строку оставить без изменений.
18. Пусть $a_i = \frac{i-1}{i+1} + \sin \frac{(i-1)^3}{i+1}$, $i=1, 2, \dots, n$, где n задано. Найти сумму всех положительных чисел a_i .
19. Определить сумму бесконечного ряда $1 - X^2/2! + X^4/4! - X^6/6! + \dots + (-1)^N X^{2N}/(2N)!$ с точностью $\epsilon > 0$ ($\epsilon \leq 0.0001$)
20. Напишите программу, запрашивающую с клавиатуры два вещественных числа и один из знаков операций $+, -, /, *$ и выводящую на экран результат соответствующего арифметического действия над этими числами.
21. Единицы длины пронумерованы следующим образом: 1 — дециметр, 2 — километр, 3 — метр, 4 — миллиметр, 5 — сантиметр. Дан номер единицы длины (целое число в диапазоне 1-5) и длина отрезка в этих единицах (вещественное число). Найти длину отрезка в метрах
22. Даны числа x, y, x_1, y_1, x_2, y_2 . Проверить истинность высказывания: «Точка с координатами (x, y) лежит внутри прямоугольника, левая верхняя вершина которого имеет координаты (x_1, y_1) , правая нижняя — (x_2, y_2) , а стороны параллельны координатным осям»
23. Функция имеет вид $J_N(x) = \left(\frac{x}{N+2}\right)^{1/2}$ при $x > 0$; Составить таблицу функции для $N=1, 2, 3, 4, 5$ при $x=0.1, 0.2, 0.3, \dots, 2$ с точностью 9 знаков после десятичной точки.
24. Дан массив размера N . Найти номер его первого локального минимума (локальный минимум — это элемент, который меньше любого из своих соседей).
25. Данна матрица размера $M \times N$. В каждом ее столбце найти количество элементов, больших среднего арифметического всех элементов этого столбца.
26. Числа Фибоначчи определяются как: $F_1 = 1, F_2 = 1, F_3 = F_2 + F_1, \dots, F_N = F_{N-1} + F_{N-2}$ при $N > 2$. Распечатать первые 30 чисел Фибоначчи в 5 колонок. Найти минимальное число Фибоначчи, кратное 7.
27. Даны действительные числа x, y, z . Вычислить u , используя функцию для вычисления \min :
- $$u = \frac{\min\left(\frac{x+y+z}{3}, xyz\right)}{1 + \min^2\left(\frac{x+y+z}{3}, xyz\right)}.$$
28. Найти площадь кольца, внутренний радиус которого равен r , а внешний — заданному числу R ($R > r$).
29. Даны x, y, z . Вычислить a, b , если: $a = \frac{2 \cos\left(x - \frac{\pi}{6}\right)}{\frac{1}{2} + \sin^2 y}, b = 1 + \frac{z^2}{3 + \frac{z^2}{5}}$.
30. Даны x, y, z . Вычислить a, b , если: $a = \frac{\sqrt{|x-1|} - \sqrt[3]{|y|}}{1 + \frac{x^2}{2} + \frac{y^2}{4}}, b = x \left(\operatorname{arctg} z + e^{-(x+3)} \right)$

Показатели и критерии оценивания планируемых результатов освоения компетенций и результатов обучения, шкала оценивания

Шкала оценивания	Формируемые компетенции	Индикатор достижения компетенции	Критерии оценивания	Уровень освоения компетенций
85 – 100 баллов	«отлично»	ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Разрабатывает и реализует эффективные алгоритмы и программы с использованием современных технологий и языков программирования	<p>Знает верно и в полном объеме: теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных.</p> <p>Умеет верно и в полном объеме: выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программиривать приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.</p>
70 – 84 баллов	«хорошо»	ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Разрабатывает и реализует эффективные алгоритмы и программы с использованием современных технологий и языков программирования	<p>Знает с незначительными замечаниями: теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных.</p> <p>Умеет с незначительными замечаниями: выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи.</p>

				дачи; программировать приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.	
50 – 69 баллов	«удовлетворительно»	ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Разрабатывает и реализует эффективные алгоритмы и программы с использованием современных технологий и языков программирования	<p>Знает на базовом уровне, с ошибками: теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных.</p> <p>Умеет на базовом уровне, с ошибками: выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной задачи; программировать приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.</p>	Базовый
менее 50 баллов	«неудовлетворительно»	ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Разрабатывает и реализует эффективные алгоритмы и программы с использованием современных технологий и языков программирования	<p>Не знает на базовом уровне: теоретические основы алгоритмизации и программирования; основные языки программирования и современные программные среды разработки информационных систем и технологий; современные методы разработки и реализации алгоритмов решения прикладных задач с использованием различных языков программирования; технологию разработки и отладки программ; синтаксис и семантику языков программирования, виды вычислительных процессов, типы данных.</p> <p>Не умеет на базовом уровне: выбирать и применять современные языки программирования для реализации алгоритмов и написания прикладных программ в зависимости от типа прикладной за-</p>	Компетенции не сформированы

				дачи; программировать приложения и создавать программные прототипы решения прикладных задач; разрабатывать программный код в современной среде программирования, анализировать и адаптировать под нужды заказчика.	
--	--	--	--	--	--