

**Министерство науки и высшего образования РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Национальный исследовательский университет «МЭИ»**

**Направление подготовки/специальность: 10.03.01 Информационная безопасность**

**Наименование образовательной программы: Организация и технология защиты информации**

**Уровень образования: высшее образование - бакалавриат**

**Форма обучения: Очная**

**Оценочные материалы  
по дисциплине  
Языки программирования**

**Москва  
2022**

## ОЦЕНОЧНЫЕ МАТЕРИАЛЫ РАЗРАБОТАЛ:

Преподаватель

(должность)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Буданова Н.А.
	Идентификатор	R5d64ed63-TrofimovaNA-be93810

(подпись)

Н.А.

Буданова

(расшифровка  
подписи)

## СОГЛАСОВАНО:

Руководитель  
образовательной  
программы

(должность, ученая степень, ученое  
звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Баронов О.Р.
	Идентификатор	R90d76356-BaronovOR-7bf8fd7e

(подпись)

О.Р. Баронов

(расшифровка  
подписи)

Заведующий  
выпускающей кафедры

(должность, ученая степень, ученое  
звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Невский А.Ю.
	Идентификатор	R4bc65573-NevskyAY-0b6e493d

(подпись)

А.Ю.

Невский

(расшифровка  
подписи)

## ОБЩАЯ ЧАСТЬ

Оценочные материалы по дисциплине предназначены для оценки: достижения обучающимися запланированных результатов обучения по дисциплине, этапа формирования запланированных компетенций и уровня освоения дисциплины.

Оценочные материалы по дисциплине включают оценочные средства для проведения мероприятий текущего контроля успеваемости и промежуточной аттестации.

Формируемые у обучающегося компетенции:

1. ОПК-4 способностью понимать значение информации в развитии современного общества, применять информационные технологии для поиска и обработки информации
2. ПК-2 способностью применять программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования для решения профессиональных задач

и включает:

**для текущего контроля успеваемости:**

Форма реализации: Защита задания

1. КМ1. Операторы ввода-вывода, вычисления (Семинар)
2. КМ2. Ветвления. Сложные условия (Семинар)
3. КМ3. Циклы (Семинар)
4. КМ4. Процедуры и функции (Семинар)
5. КМ5. Одномерные массивы (Семинар)
6. КМ6. Матрицы (Семинар)
7. КМ7. Символьные строки (Семинар)
8. КМ8. Алгоритмы сортировки (Семинар)

## БРС дисциплины

4 семестр

Раздел дисциплины	Веса контрольных мероприятий, %								
	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4	КМ-5	КМ-6	КМ-7	КМ-8
	Срок КМ:	2	4	6	8	10	12	14	16
Основы создания программ									
Тема 1. Основы технологии разработки программ	+								
Тема 2. Диалоговые программы	+								
Тема 3. Основные операторы, вычисления.	+								
Ветвления.									
Тема 1. Условный оператор			+						

Тема 2. Вложенный условный оператор		+						
Тема 3. Логические переменные и сложные условия		+						
Циклы								
Тема 1. Циклы с постусловием			+					
Тема 2. Циклы с предусловием			+					
Тема 3. Параметрический цикл.			+					
Процедуры и функции								
Тема 1. Создание процедур.				+				
Тема 2. Создание функций.				+				
Одномерные массивы								
Тема 1. Создание статического массива					+			
Тема 2. Алгоритмы работы с одномерными массивами					+			
Тема 3. Динамический массив					+			
Матрицы								
Тема 1. Основные понятия по теме Матрицы						+		
Тема 2. Основные алгоритмы работы с матрицами						+		
Символьные строки								
Тема 1. Основные понятия по теме Символьные строки							+	
Тема 2. Методы работы со строками							+	
Алгоритмы сортировки								
Тема 1. Простые алгоритмы сортировки								+
Тема 2. Быстрые алгоритмы сортировки								+
Вес КМ:	15	10	10	10	15	15	10	15

\$Общая часть/Для промежуточной аттестации\$

## СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

### *I. Оценочные средства для оценки запланированных результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций*

Индекс компетенции	Индикатор	Запланированные результаты обучения по дисциплине	Контрольная точка
ОПК-4	ОПК-4(Компетенция)	<p>Знать: Методы разработки и внедрения прикладного программного обеспечения. Особенности создания диалоговых программ. Знать основные операторы.</p> <p>Уметь: Создавать диалоговые программы. Отлаживать и тестировать программы на языках программирования С и С++.</p>	КМ1. Операторы ввода-вывода, вычисления (Семинар)
ПК-2	ПК-2(Компетенция)	<p>Знать: Принципы программирования на языках программирования С и С++ . Особенности циклов. Основные алгоритмы сортировки Структуру программы на языках программирования С и С++. Особенности</p>	<p>КМ2. Ветвления. Сложные условия (Семинар) КМ3. Циклы (Семинар) КМ4. Процедуры и функции (Семинар) КМ5. Одномерные массивы (Семинар) КМ6. Матрицы (Семинар) КМ7. Символьные строки (Семинар) КМ8. Алгоритмы сортировки (Семинар)</p>

		<p>работы условного оператора. Вложенного условного оператора.</p> <p>Правила создания подпрограмм на языке C и C++.</p> <p>Методы работы со строками на языке C и C++</p> <p>Особенности работы с матрицами на языке C и C++.</p> <p>Особенности работы с одномерными массивами на языке C и C++.</p> <p>Уметь:</p> <p>Реализовывать подпрограммы на языке C и C++</p> <p>Создавать строковые переменные и применять основные методы работы со строками</p> <p>Использовать различные виды циклов для различных практических задач</p> <p>Работать с одномерными массивами</p> <p>Работать с матрицами</p> <p>Реализовывать простые методы сортировки</p> <p>Использовать условный оператор для решения</p>	
--	--	--	--

		практических задач	
--	--	--------------------	--

## II. Содержание оценочных средств. Шкала и критерии оценивания

### КМ-1. КМ1. Операторы ввода-вывода, вычисления

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС:** 15

**Процедура проведения контрольного мероприятия:** Студент предоставляет отчет по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

#### Краткое содержание задания:

Пример задания. Написать программу, которая меняет местами значение двух переменных в памяти без использования дополнительной переменной

#### Контрольные вопросы/задания:

Знать: Методы разработки и внедрения прикладного программного обеспечения. Особенности создания диалоговых программ. Знать основные операторы.	1.Пример вопроса. Чем характеризуется ячейка памяти?
Уметь: Создавать диалоговые программы. Отлаживать и тестировать программы на языках программирования С и С++.	1.Пример вопроса. Как осуществить ввод в программе: от пользователя, случайным образом и с помощью инициализации при объявлении переменных?

#### Описание шкалы оценивания:

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания:* Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания:* Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

*Оценка: 3*

*Нижний порог выполнения задания в процентах: 50*

*Описание характеристики выполнения знания:* Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

### КМ-2. КМ2. Ветвления. Сложные условия

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС:** 10



**Процедура проведения контрольного мероприятия:** Студент предоставляет отчёт по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

**Краткое содержание задания:**

Пример задания. Написать программу, которая вводит координаты точки на плоскости и определяет, попала ли эта точка в заштрихованную область.

**Контрольные вопросы/задания:**

Знать: Структуру программы на языках программирования С и С++. Особенности работы условного оператора. Вложенного условного оператора.	1.Пример вопроса. Какой приоритет выполнения операций при составлении сложных условий?
Уметь: Использовать условный оператор для решения практических задач	1.Пример вопроса. Как изменить программу, чтобы определялось попадание точки только на грануцу области?

**Описание шкалы оценивания:**

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания:* Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания:* Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

*Оценка: 3*

*Нижний порог выполнения задания в процентах: 50*

*Описание характеристики выполнения знания:* Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

**КМ-3. КМ3. Циклы**

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС:** 10

**Процедура проведения контрольного мероприятия:** Студент предоставляет отчёт по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

**Краткое содержание задания:**

Пример задания. Написать программу, которая проверяет правильно ли человек ввёл год рождения. Допустимым считать значения от 1900 до 2021.

**Контрольные вопросы/задания:**

Знать: Принципы программирования на языках	1.Пример вопроса. Нарисуйте блок-схему цикла по переменной?
--	---

программирования С и С++ . Особенности циклов.	
Уметь: Использовать различные виды циклов для различных практических задач	1.Пример вопроса. Как можно реализовать цикл с постусловием, без использования конструкции do while?

#### Описание шкалы оценивания:

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно*

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач*

*Оценка: 3*

*Нижний порог выполнения задания в процентах: 50*

*Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено*

#### КМ-4. КМ4. Процедуры и функции

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС:** 10

**Процедура проведения контрольного мероприятия:** Студент предоставляет отчет по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

#### Краткое содержание задания:

Пример задания. Написать процедуру с параметрами, которая меняет значения переменных местами.

#### Контрольные вопросы/задания:

Знать: Правила создания подпрограмм на языке С и С++.	1.Пример вопроса. Чем процедура отличается от функции?
Уметь: Реализовывать подпрограммы на языке С и С++	1.Пример вопроса. Что произойдет, если при передачи параметров по ссылке убрать знак &?

#### Описание шкалы оценивания:

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно*

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач*

*Оценка: 3*

*Нижний порог выполнения задания в процентах: 50*  
*Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено*

### **КМ-5. КМ5. Одномерные массивы**

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС: 15**

**Процедура проведения контрольного мероприятия:** Студент предоставляет отчёт по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

**Краткое содержание задания:**

Пример задания. Написать программу, которая находит максимальный элемент в массиве

**Контрольные вопросы/задания:**

Знать: Особенности работы с одномерными массивами на языке С и С++.	1.Пример вопроса. Какие существуют способы задания элементов массива?
Уметь: Работать с одномерными массивами	1.Пример вопроса. Как изменить программу, чтобы найти минимальный элемент массива?

**Описание шкалы оценивания:**

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно*

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач*

*Оценка: 3*

*Нижний порог выполнения задания в процентах: 50*

*Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено*

### **КМ-6. КМ6. Матрицы**

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС: 15**

**Процедура проведения контрольного мероприятия:** Студент предоставляет отчёт по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

**Краткое содержание задания:**

Пример задания. Написать программу, которая меняет местами первую и последнюю строки матрицы.

**Контрольные вопросы/задания:**

Знать: Особенности работы с матрицами на языке C и C++.	1.Пример вопроса. Как матрица хранится в памяти компьютера?
Уметь: Работать с матрицами	1.Пример вопроса. Как поменять размерность матрицы в программе?

**Описание шкалы оценивания:**

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно*

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач*

*Оценка: 3*

*Нижний порог выполнения задания в процентах: 50*

*Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено*

**КМ-7. КМ7. Символьные строки**

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС:** 10

**Процедура проведения контрольного мероприятия:** Студент предоставляет отчет по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

**Краткое содержание задания:**

Пример задания. Написать программу, которая удаляет двойные пробелы из строки

**Контрольные вопросы/задания:**

Знать: Методы работы со строками на языке C и C++	1.Пример задания. Чем символьные строки отличаются от массивов?
Уметь: Создавать строковые переменные и применять основные методы работы со строками	1.Пример задания. Как в программе осуществить вывод только первого слова из строки?

**Описание шкалы оценивания:**

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно*

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания:* Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

*Оценка:* 3

*Нижний порог выполнения задания в процентах:* 50

*Описание характеристики выполнения знания:* Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

### **КМ-8. КМ8. Алгоритмы сортировки**

**Формы реализации:** Защита задания

**Тип контрольного мероприятия:** Семинар

**Вес контрольного мероприятия в БРС:** 15

**Процедура проведения контрольного мероприятия:** Студент предоставляет отчёт по заданиям. В ходе беседы со студентом выявляется на сколько хорошо студент овладел понятийно-терминологическим аппаратом, способен ли применять теорию к практическим заданиям.

#### **Краткое содержание задания:**

Пример задания. Написать программу реализующую метод сортировки пузырьком?

#### **Контрольные вопросы/задания:**

Знать: Основные алгоритмы сортировки	1.Пример задания. Какие существуют методы сортировки?
Уметь: Реализовывать простые методы сортировки	1.Пример задания. Исправить программу, чтобы сортировка проводилась методом камня.

#### **Описание шкалы оценивания:**

*Оценка:* 5

*Нижний порог выполнения задания в процентах:* 70

*Описание характеристики выполнения знания:* Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

*Оценка:* 4

*Нижний порог выполнения задания в процентах:* 60

*Описание характеристики выполнения знания:* Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

*Оценка:* 3

*Нижний порог выполнения задания в процентах:* 50

*Описание характеристики выполнения знания:* Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

# СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

4 семестр

Форма промежуточной аттестации: Экзамен

Пример билета

НИУ МЭИ	<b>ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1</b> Кафедра <i>Безопасности и информационных технологий</i> Дисциплина «Языки программирования»	Утверждаю: Зав. каф. БИТ А.Ю.Невский
		Протокол №6 «19» 05 2021 г.
1. Алгоритм. Исполнитель. Система команд исполнителя. Программа. 2. Двоичный поиск. 3. Задача		

## Процедура проведения

Студент вытягивает билет. На подготовку отводится 40 минут. Студент отвечает на вопросы билета. Решает задачу с использованием ПК.

## *1. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины*

### **1. Компетенция/Индикатор: ОПК-4(Компетенция)**

#### **Вопросы, задания**

- 1.Алгоритм. Исполнитель. Система команд исполнителя. Программа.
- 2.Алгоритм. Свойства алгоритма. Обязательные.
- 3.Алгоритм. Свойства алгоритма. Необязательные.
4. Алгоритм. Способы описания алгоритма.
5. Описание алгоритма с помощью Блок-схемы (основные блоки).
- 6.Трассировка. Трансляция. Компиляция. Интерпретация.
- 7.Классификация языков программирования по способу трансляции.
- 8.Среда программирования. Основные компоненты. Направления программирования.
- 9.Этапы прохождения программы, требующей компиляции.
- 10.Этапы создания программ. Постановка задачи. Построение модели. Разработка алгоритма. Кодирование.
- 11.Этапы создания программ. Отладка. Тестирование. Документирование. Внедрение и сопровождение.
- 12.Структура простейшей программы на любом алгоритмическом языке. Что необходимо знать для написания простейшей программы.
- 13.Переменные. Имена переменных. Объявление переменных. Что определяет тип переменных.
- 14.Ячейка памяти (определение, чем характеризуется, свойства).
15. Базовые алгоритмические структуры. Принцип Дейкстры.
16. Метод нисходящего проектирования.

## Материалы для проверки остаточных знаний

### 1. Этапы создания программ.

Верный ответ: Вопрос 1. Этапы создания программ. Постановка задачи Построение модели Разработка алгоритма и способа представления данных Кодирование Отладка Тестирование Документирование внедрение и сопровождение

### 2. Метод нисходящего проектирования.

Верный ответ: Разбиение алгоритма на части и установлении между ними связей. При установлении связей очень важно, чтобы каждая часть имела один вход и один выход, так что нисходящее проектирование успешно сочетается с использованием базовых структур алгоритмов. Каждая часть в свою очередь разбивается на части, и процесс повторяется. Можно сказать, что нисходящее проектирование алгоритма состоит в иерархической последовательной разработке алгоритма от сложного к простому.

### 3. Алгоритм. Способы записи алгоритма.

Верный ответ: Словесная запись Запись «по шагам» Блок-схема алгоритма Программа на псевдокоде (Псевдокод – алгоритм на русском языке с элементами языка программирования) Машинная программа Программа на алгоритмическом языке

### 4. Алгоритм. Свойства алгоритма. Обязательные.

Верный ответ: Дискретность - алгоритм состоит из отдельных команд, каждая из которых выполняется ограниченное время. Понятность - алгоритм содержит только команды, входящие в систему команд исполнителя, для которого он предназначен. Определённость - при каждом выполнении с одними и теми же данными должен быть получен один и тот же результат.

### 5. Алгоритм. Свойства алгоритма. Необязательные.

Верный ответ: Конечность (результативность) — для корректного набора данных алгоритм должен заканчиваться с некоторым результатом (не заикливаться). Корректность — для допустимых исходных данных алгоритм должен приводить к правильному результату. Массовость — алгоритм можно использовать для решения множества однотипных задач с различными исходными данными (решение «в буквах»).

### 6. Алгоритм. Исполнитель. Система команд исполнителя. Программа.

Верный ответ: Алгоритм - это совокупность правил для решения некоторого класса задач, последовательное применение которых за конечное число шагов приводит к определенному результату. Исполнитель – это устройство или одушевленное существо (человек), способное понять и выполнить команды, составляющие алгоритм. Формальные исполнители: не понимают (и не могут понять) смысл команд. Полный набор команд исполнителя называется системой команд исполнителя (СМК)

## 2. Компетенция/Индикатор: ПК-2(Компетенция)

### Вопросы, задания

1. Классификация данных по функциональному назначению, по структуре, по типу.
2. Базовые алгоритмические структуры. Следование. Разветвление.
3. Базовые алгоритмические структуры. Цикл с предусловием. Цикл с постусловием.
4. Базовые алгоритмические структуры. Цикл с переменной (или параметрический цикл).
5. Операторы ввода-вывода в языке C и C++ (форматы вывода).

## Материалы для проверки остаточных знаний

### 1. C++. Случайные и псевдослучайные числа.

Верный ответ: Для использования случайных чисел в C++ нужно подключить библиотеку `random`. Функция `rand` возвращает случайное значение на отрезке

[0;RAND\_MAX]  $k = \text{rand}() \% N$ ; - операция остаток от деления позволяет найти случайное число на отрезке [0;N-1] Для задания случайного числа на отрезке [a,b] нужно воспользоваться формулой  $k = a + \text{rand}() \% (b - a + 1)$ ; Для изменения зерна (чтобы каждый раз появлялось новое псевдослучайное число) нужно использовать функцию `srand`.

2. C++. Условный оператор. Полная и неполная форма записи.

Верный ответ: Полная форма `if (условие) оператор 1; else оператор 2;` В неполной форме ветвь "нет" - пустая, следовательно `else` и оператор 2 отсутствуют.

3. C++. Сложные условия. Приоритет.

Верный ответ: `&&` - and - «И» `||` - or - «ИЛИ» `!` - not - «НЕ» Приоритет : Операции в скобках Операции НЕ Арифметические операции (+, -, \*, /, %) Операции сравнения (<, >, <=, >=, ==, !=) `&&` («И») `||` («ИЛИ»)

4. C++. Множественный выбор.

Верный ответ: `if (m == 1) cout << "текст 1"; if (m == 2) cout << "текст 2"; ... if (m == 12) cout << "текст 3";` можно реализовать с помощью специального оператора множественного выбора: `switch ( m ) { case 1: cout << "текст 1"; break; case 2: cout << "текст 2"; break; ... case 12: cout << "текст 3"; break; default: cout << "ошибка"; }`

5. Символьные строки. Определение. Объявление. Ввод строки.

Верный ответ: Символьная строка – это последовательность символов, которая рассматривается как единый объект. `string str1;` Присваивание: `string str2="Привет"; str2="Здравствуй"; string str3(5,'f'); str3= string(10,'+');`

6. Определение длины строки. Сравнение строк.

Верный ответ: `string` – объект -> свойства и методы. Метод – функция для обработки данных объекта. `int n=s.size();`

7. Объединение строк. Обращение к символу.

Верный ответ: Оператор «+» `string str1="Иванов"; string str2="Кузнецов"; string str3=str1+" и "+str2+" друзья!";` Каждый символ строки имеет свой индекс, нумерация начинается с нуля.

8. Перебор всех символов в строке.

Верный ответ: Используем цикл по переменной `string str1="I love lectures"; for(int i=0;i<str1.size();i++){ cout<<"str1["<<i<<"]"<<str1[i]<<endl;}` //int(str1[i])-коды символов

9. Подстроки. Удаление и вставка.

Верный ответ: Метод выделяющий часть строки `.substr(индекс символа, количество символов)` `string str1="I love lectures",str2; str2=str1.substr(2,4); cout<<str2<<endl;` строка.`erase(индекс первого удаляемого, сколько удалить)` строка.`clear();` - полностью очищает строку /\* строка.`empty()` – истина или строка.`size()==0` \*/ строка.`insert(индекс начала вставки, фрагмент для вставки)`

10. Поиск в символьных строках.

Верный ответ: строка.`find(строка/символ, индекс с какого искать)` Возвращает индекс первого найденного символа; При поиске подстроки- индекс первого вхождения подстроки; Если не найден – `string::npos` `int(string::npos)` – это -1; строка.`rfind()` – от англ. reverse find – поиск последнего вхождения образца в строку.

11. Преобразование строки в число и числа в строку.

Верный ответ: `#include <iostream> #include <string> using namespace std; int main(){ string test = "45.123hghjg"; int myint = stoi(test); float myfloat = stof(test); cout<<myint<<" "<<myfloat; return 0; }` `int myint=123; float myfloat=123.5898; string temp1=to_string(myint); string temp2=to_string(myfloat); cout<<myint<<" "<<myfloat;`

12. Одномерные массивы. Перебор элементов. Ввод и вывод массивов.



Верный ответ: `const int N = 5; int A[N]={}; //все 0 Цикл с переменной: for( i = 0; i < N; i++) { // обработать A[i] (A[i]=i) } for( int& x: A ) X=1; //заполнение массива единицами cout<<A for ( i = 0; i < N; i++) cout << A[i] << " "; for( int x: A ) cout<<x<< " "; or ( i = 0; i < N; i++) { // cout << "A[" << i << "]"="; cin >> A[i]; } for( int& x: A ) cin>>x;`

13. Одномерные массивы. Определение. Объявление. Заполнение. Обращение.

Верный ответ: Массив – это группа переменных одного типа, расположенных в памяти рядом (в соседних ячейках) и имеющих общее имя. Каждая ячейка в массиве имеет уникальный номер (индекс). выделять память записывать данные в нужную ячейку читать данные из ячейки Объявление массива: `int A[5]; double V[8]; char Sym[80]; const int MAX_EL = 10; int A[MAX_EL] int A[5]={2,4,7,1,53}; int B[25]={}; int C[]={2,44,55,3} string students[]={ "Иванов, Петров, Сидоров"}` Индексом может быть также значение арифметического выражения – целое число Пример: `int B[]={3,5,4,2,8,9}; int i=1; cout<<B[i]<<B[i+1]<<B[2*i+2]<<B[B[3*i]] //1,2,4,2`

## **II. Описание шкалы оценивания**

*Оценка: 5*

*Нижний порог выполнения задания в процентах: 70*

*Описание характеристики выполнения знания: Работа выполнена в рамках "продвинутого" уровня. Ответы даны верно, четко сформулированные особенности практических решений*

*Оценка: 4*

*Нижний порог выполнения задания в процентах: 60*

*Описание характеристики выполнения знания: Работа выполнена в рамках "базового" уровня. Большинство ответов даны верно. В части материала есть незначительные недостатки*

*Оценка: 3*

*Нижний порог выполнения задания в процентах: 50*

*Описание характеристики выполнения знания: Работа выполнена в рамках "порогового" уровня. Основная часть задания выполнена верно. на вопросы углубленного уровня*

## **III. Правила выставления итоговой оценки по курсу**

Оценка определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ» на основании семестровой и экзаменационной составляющих.