

**ПЕРВОЕ ВЫСШЕЕ ТЕХНИЧЕСКОЕ УЧЕБНОЕ ЗАВЕДЕНИЕ РОССИИ**



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОРНЫЙ УНИВЕРСИТЕТ**

**СОГЛАСОВАНО**

**УТВЕРЖДАЮ**

---

**Руководитель ОПОП ВО**  
профессор  
**К.В. Гоголинский**

---

**Проректор по образовательной**  
деятельности  
доцент **Д.Г. Петраков**

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
**ТЕОРИЯ И ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ**

**Уровень высшего образования:** Бакалавриат

**Направление подготовки:** 27.03.01 Стандартизация и метрология

**Направленность (профиль):** Метрология и метрологическое обеспечение

**Квалификация выпускника:** Бакалавр

**Форма обучения:** Очная

**Составитель:** Ассистент А.В. Кондратьев

Санкт-Петербург

**Рабочая программа дисциплины «Теория и технология программирования» разработана:**

- в соответствии с требованиями ФГОС ВО – бакалавриат по направлению подготовки «27.03.01 Стандартизация и метрология», утвержденного приказом Минобрнауки России № 901 от 07.08.2020 г.;

- на основании учебного плана бакалавриата по направлению подготовки «27.03.01 Стандартизация и метрология» направленность (профиль) «Метрология и метрологическое обеспечение».

Составитель \_\_\_\_\_ к.т.н., ассистент Кондратьев А.В.

**Рабочая программа рассмотрена и одобрена** на заседании кафедры метрологии, приборостроения и управления качеством от 03.02.2021 г., протокол № 10.

Заведующий кафедрой \_\_\_\_\_ д.т.н., профессор К.В. Гоголинский

**Рабочая программа согласована:**

Начальник управления учебно-методического обеспечения образовательного процесса \_\_\_\_\_ к.т.н. Иванова П.В.

## 1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

*Цель* – приобретение базовых знаний в области разработки программного обеспечения, классификации программного обеспечения, умения использовать языки программирования низкого и высокого уровня для задач стандартизации и метрологии.

*Задачи:*

- ознакомление с этапами разработки и моделями жизненного цикла программного обеспечения;
- изучение теории программирования;
- изучение классификации языков программирования, а также критериев их выбора для решения профессиональных задач;
- получение практических навыков программирования на языках низкого и высокого уровня;
- освоение процессов разработки алгоритмов, тестирования, отладки, анализа результатов и дальнейшего сопровождения программного обеспечения.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Теория и технология программирования» относится к части, формируемой участниками образовательных отношений, раздел «ФТД. Факультативные дисциплины» основной профессиональной образовательной программы по направлению подготовки

«27.03.01 Стандартизация и метрология», направленность (профиль) «Метрология и метрологическое обеспечение» и изучается в 8 семестре.

Предшествующими курсами, на которых непосредственно базируется дисциплина «Теория и технология программирования» являются «Введение в информационные технологии», «Теория вероятностей и математическая статистика», «Электроника и микропроцессорная техника».

Особенностью дисциплины является ее исключительная важность для успешной деятельности будущих специалистов в области стандартизации и метрологии в рамках концепции «Индустрия 4.0», направленность на получение практических знаний в области программирования, а также тесная связь с существующими профессиональными задачами.

## 3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Процесс изучения дисциплины «Теория и технология программирования» направлен на формирование следующих компетенций:

Формируемые компетенции		Код и наименование индикатора достижения компетенции
Содержание компетенции	Код компетенции	
Способен участвовать в планировании работ по метрологическому обеспечению, систематически проверять соответствие применяемых нормативных документов действующим правовым актам и тенденциям развития технического	ПКС-1	ПКС-1.2. Умеет проводить анализ применяемой на предприятии нормативно-технической документации в области метрологического обеспечения.

Формируемые компетенции		Код и наименование индикатора достижения компетенции
Содержание компетенции	Код компетенции	
регулирования		
Способен определять номенклатуру измеряемых и контролируемых параметров продукции и технологических процессов, устанавливать нормы точности измерений и достоверности контроля, выбирать средства измерений и контроля, разрабатывать локальные поверочные схемы и проводить поверку, калибровку средств измерений	ПКС-3	ПКС-3.4. Умеет выбирать требуемые нормы точности измерений и достоверности контроля  ПКС-3.5. Владеет навыками обработки данных и оценки точности измерений

#### 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

##### 4.1. Объем дисциплины и виды учебной работы

Общая трудоёмкость учебной дисциплины составляет 2 зачётных единицы, 72 ак. часа.

Вид учебной работы	Всего ак. часов	Ак. часы по семестрам
		72
<b>Аудиторная работа, в том числе:</b>	<b>33</b>	<b>33</b>
Лекции (Л)	11	11
Практические занятия (ПЗ)	-	-
Лабораторные работы (ЛР)	22	22
<b>Самостоятельная работа студентов (СРС), в том числе:</b>	<b>39</b>	<b>39</b>
Подготовка к лабораторным занятиям	16	16
Подготовка к зачету / дифф. зачету	9	9
Работа в библиотеке	14	14
<b>Промежуточная аттестация – дифф. зачет</b>	<b>ДЗ</b>	<b>ДЗ</b>
<b>Общая трудоёмкость дисциплины</b>		
	<b>ак. час.</b>	<b>72</b>
	<b>зач. ед.</b>	<b>2</b>

##### 4.2. Содержание дисциплины

Учебным планом предусмотрены: лекции, практические занятия и самостоятельная работа.

#### 4.2.1. Разделы дисциплины и виды занятий

Наименование разделов	Виды занятий				
	Всего ак. часов	Лекции	Практические занятия	Лабораторные работы	Самостоятельная работа студента,
Раздел 1 «Теория программирования»		3	-	-	5
Раздел 2 «Технология программирования»		4	-	-	17
Раздел 3 «Программирование для математических вычислений»		4	-	-	17
<b>Итого:</b>	<b>72</b>	<b>11</b>	<b>-</b>	<b>22</b>	<b>39</b>

#### 4.2.2. Содержание разделов дисциплины

№ п/п	Наименование раздела дисциплины	Содержание лекционных занятий	Трудоемкость в ак. Часах
1	Раздел 1. «Теория программирования»	Введение. Цели и задачи дисциплины. Этапы развития технологий программирования: стихийное программирование, структурное, модульное программирование, объектный подход, компонентный подход.	3
2	Раздел 2 «Технология программирования»	Введение в анализ алгоритмов. Рекурсия. Сортировки. Динамические структуры данных. Поиск в структурах данных. Графы и деревья. Алгоритмы на графах и деревьях.	4
3	Раздел 3 «Программирование для математических вычислений»	Вычислительные задачи, возникающие в процессе решения задач стандартизации и метрологии.	4
<b>Итого:</b>			<b>11</b>

#### 4.2.3. Лабораторные занятия

№ п/п	Разделы	Тематика практических занятий	Трудоемкость в ак. часах
1	Раздел 1	Работа в различных средах программирования.	4
2	Раздел 2	Анализ сложности алгоритмов. Алгоритмы: сортировки, поиска. Алгоритмы на графах и деревьях.	6
3	Раздел 3	Функции и логические выражения. Матрицы. Решение систем линейных уравнений. Построение графиков. Интегрирование и дифференцирование.	12
<b>Итого:</b>			<b>22</b>

#### 4.2.4. Практические работы

Практические работы не предусмотрены.

#### 4.2.5. Курсовые работы (проекты)

Курсовые работы (проекты) не предусмотрены.

### 5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В ходе обучения применяются:

**Лекции**, которые являются одним из важнейших видов учебных занятий и составляют основу теоретической подготовки обучающихся. Цели лекционных занятий:

-дать систематизированные научные знания по дисциплине, акцентировать внимание на наиболее сложных вопросах дисциплины;

-стимулировать активную познавательную деятельность обучающихся, способствовать формированию их творческого мышления.

**Лабораторные занятия.** Цели лабораторных занятий:

-углубить и закрепить знания, полученные на лекциях и в процессе самостоятельной работы обучающихся с учебной и научной литературой;

Главным содержанием этого вида учебных занятий является работа каждого обучающегося по овладению практическими умениями и навыками профессиональной деятельности.

**Консультации** (текущая консультация, накануне *дифф. зачета*) является одной из форм руководства учебной работой обучающихся и оказания им помощи в самостоятельном изучении материала дисциплины, в ликвидации имеющихся пробелов в знаниях, задолженностей по текущим занятиям, в подготовке письменных работ (проектов).

Текущие консультации проводятся преподавателем, ведущим занятия в учебной группе, научным руководителем и носят как индивидуальный, так и групповой характер.

**Самостоятельная работа обучающихся** направлена на углубление и закрепление знаний, полученных на лекциях и других занятиях, выработку навыков самостоятельного активного приобретения новых, дополнительных знаний, подготовку к предстоящим учебным занятиям и промежуточному контролю.

### 6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

#### 6.1. *Оценочные средства для самостоятельной работы и текущего контроля успеваемости*

##### Раздел 1. Теория программирования

1. Что такое «технология программирования»? Какие инструкции она в себя включает? Опишите структуру технологической операции.

2. Каковы основные этапы развития технологии программирования? «Стихийное» программирование, его суть, достоинства и недостатки.

3. Каковы основные этапы развития технологии программирования? Структурное программирование, его суть, достоинства и недостатки.

4. Каковы основные этапы развития технологии программирования? Объектно-ориентированное программирование, его суть, достоинства и недостатки.

5. Выделите семь принципов объектно-ориентированного подхода.

##### Раздел 2 «Технология программирования»

1. Сформулируйте эксплуатационные требования к программным продуктам.

2. Изучите следующую нормативную документацию: стандарты разработки, стандарт проектирования, стандарт оформления проектной документации, стандарт интерфейса пользователя.

3. Перечислите и опишите основные алгоритмические конструкции.

4. Изучите основные алгоритмы сортировки.

5. Изучите методы поиска в графе. Поиск в ширину. Поиск в глубину.

### Раздел 3 «Программирование для математических вычислений»

1. Операции над матрицами (по вариантам).
2. Решение систем линейных уравнений (по вариантам).
3. Вычисление интегралов (по вариантам).
4. Решение дифференциальных уравнений (по вариантам).
5. Построение графиков (по вариантам).

#### 6.2. Оценочные средства для проведения промежуточной аттестации (дифф. зачет)

##### 6.2.1. Примерный перечень вопросов/заданий к дифф. зачету (по дисциплине):

1. Назовите различия между классом и структурой в C ++.
2. Что делает функция Friend в C ++?
3. В чем разница между вектором и массивом?
4. Каковы спецификаторы доступа в C ++?
5. Что такое наследование в объектно-ориентированном программировании?
6. В чем разница между абстрактными и интерфейсными классами?
7. В чем различия между инкапсуляцией и абстракцией?
8. В чем разница между открытым и закрытым контуром?
9. Что такое концепция OOPS и как она реализована в C ++?
10. В чем разница между ссылкой и указателем?
11. Что такое виртуальные функции?
12. Что такое виртуальные конструкторы / деструкторы?
13. В чем разница между обычным указателем и указателем void?
14. Что такое многопоточное программирование?
15. Как C ++ работает с конструкторами класса и его дочернего класса?
16. Зачем нужен виртуальный деструктор?
17. Для каких целей применяется ключевое слово const?
18. Сколько в памяти занимает произвольная структура?
19. Для чего нужен аллокатор и как создать свой собственный аллокатор?
20. В чём суть множественного наследования?

##### 6.2.2. Примерные тестовые задания к дифф. зачету

Вариант №1

№ п/п	Вопрос	Варианты ответа
1.	Укажите тип данных, которое определяет служебное слово struct:	<ol style="list-style-type: none"><li>1. Массив данных с различной структурой;</li><li>2. Тип функций, которые могут иметь различную структуру параметров;</li><li>3. Тип данных, которые могут менять свою структуру;</li><li>4. Составной объект, к которому могут входить элементы различных типов.</li></ol>

2.	Укажите типы доступа, возможные для элементов структур:	<ol style="list-style-type: none"> <li>1. Любые;</li> <li>2. Только закрытые;</li> <li>3. Только открытые;</li> <li>4. Только защищенные.</li> </ol>
3.	Определите, для чего предназначены конструкторы класса:	<ol style="list-style-type: none"> <li>1. Для конструирования класса;</li> <li>2. Для инициализации объектов класса;</li> <li>3. Для выделения памяти объектам класса;</li> <li>4. Для объявления объектов класса.</li> </ol>
4.	Что такое производный класс?	<ol style="list-style-type: none"> <li>1. Класс, построенный на основе другого класса;</li> <li>2. Класс, на основе которого построен другой класс;</li> <li>3. Класс, на основе которого построено ни одного класса;</li> <li>4. Класс, у которого кроме элементов являются методы.</li> </ol>
5.	Каких спецификаторов доступа не существуют?	<ol style="list-style-type: none"> <li>1. Public;</li> <li>2. Protected;</li> <li>3. Private;</li> <li>4. Open.</li> </ol>
6.	Имеет ли доступ производный класс к закрытым элементам своего базового класса?	<ol style="list-style-type: none"> <li>1. Имеет;</li> <li>2. Нет;</li> <li>3. Имеет при открытом наследовании;</li> <li>4. Имеет при отсутствии спецификатора наследования.</li> </ol>
7.	Что такое множественное наследование?	<ol style="list-style-type: none"> <li>1. Наследование нескольких производных классов от одного базового;</li> <li>2. Наследование производного класса от нескольких базовых;</li> <li>3. Наследование производного класса с спецификатором <code>mapu</code>;</li> <li>4. Наследование производного класса от класса, который является производным от другого класса.</li> </ol>
8.	С какого символа начинается запись директивы препроцессора?	<ol style="list-style-type: none"> <li>1. @;</li> <li>2. #;</li> <li>3. &lt;;</li> <li>4. ».</li> </ol>
9.	Укажите оператора выбора в языке C++:	<ol style="list-style-type: none"> <li>1. CASE;</li> <li>2. choice;</li> <li>3. switch ... case;</li> <li>4. default.</li> </ol>
10.	В сложных выражениях последовательность выполнения операций определяется...	<ol style="list-style-type: none"> <li>1. Только приоритетом операций;</li> <li>2. Только скобками;</li> <li>3. Скобками, приоритетом операций, а при одинаковом приоритете ассоциативностью операций;</li> <li>4. Только ассоциативностью операций.</li> </ol>



11.	В чем разница между фактическими и формальными параметрами функций?	<ol style="list-style-type: none"> <li>1. Формальные параметры могут использоваться только вне тела функции, а фактические – используются как вне функции, так и внутри нее;</li> <li>2. Нет разницы, это одни и те же параметры;</li> <li>3. И формальные, и фактические параметры используются вне тела функции;</li> <li>4. Формальные параметры определены в теле функции, а фактические – значение, с которыми функция вызывается.</li> </ol>
12.	В каком файле заголовков определен объект cout?	<ol style="list-style-type: none"> <li>1. Stream.h;</li> <li>2. Sysutils.hpp;</li> <li>3. Cout.h;</li> <li>4. Iostream.h.</li> </ol>
13.	Для чего предназначен оператор continue в языке C++?	<ol style="list-style-type: none"> <li>1. Пропускает остаток тела цикла и переходит к следующей итерации;</li> <li>2. Пропускает цикл и переходит к следующему оператору в теле программы;</li> <li>3. Определяет условие продолжения цикла;</li> <li>4. Продолжает выполнение текущей итерации цикла.</li> </ol>
14.	Что выполняет операция ++ в языке C++?	<ol style="list-style-type: none"> <li>1. Уменьшает значение операнда на единицу;</li> <li>2. Уменьшает значение операнда на два;</li> <li>3. Увеличивает значение операнда на два;</li> <li>4. Увеличивает значение операнда на единицу.</li> </ol>
15.	Выберите правильный вариант объявления константной переменной в C++, где type – тип данных в C++; variable – имя переменной; value – константные значения:	<ol style="list-style-type: none"> <li>1. const type variable = value;</li> <li>2. const type variable = value;</li> <li>3. const variable = value;</li> <li>4. type const variable = value.</li> </ol>
16.	В каком случае выражение C++ будет исчисляться быстрее?	<ol style="list-style-type: none"> <li>1. <math>X = X + Y</math>;</li> <li>2. <math>X += Y</math>;</li> <li>3. <math>X = Y + X</math>;</li> <li>4. Все три выражения будут выполнены с одинаковой скоростью.</li> </ol>
17.	В программе на языке C++ есть два объявления переменных <code>int qwerty; int QWERTY</code> . Какое из утверждений верно?	<ol style="list-style-type: none"> <li>1. Такие имена переменных недопустимы;</li> <li>2. Объявления правильные;</li> <li>3. Такие объявления недопустимы, потому что мы пытаемся создать две переменные с одинаковыми идентификаторами;</li> <li>4. Переменные описываются не по такому принципу.</li> </ol>
18.	Зачем в C++ используют оператор return?	<ol style="list-style-type: none"> <li>1. Чтобы задержать работу программы;</li> <li>2. Функция, в которой он содержится, завершает свое выполнение и управление возвращается в то место программы, из которого вызывалась данная функция;</li> <li>3. Чтобы организовать цикл;</li> <li>4. Чтобы ввести в программу новые значения.</li> </ol>

19.	В одном из следующих строк, записанных на языке C++, выполняется обращение к седьмому элементу массива, размер массива равен 10?	1. mas (7); 2. mas [6]; 3. mas 8; 4. mas [7].
20.	В программе на языке C++ объявлены такие переменные int x, y. Выражение позволяет вычислить остаток от деления этих переменных?	1. x% y; 2. x div y; 3. x mod y; 4. x / y;

## Вариант №2

№ п/п	Вопрос	Варианты ответа
1.	С помощью чего реализуется принцип полиморфизма в C++?	1. Наличия множественного наследования; 2. Наличия виртуальных методов; 3. Использование виртуального наследования; 4. Наличия абстрактных классов.
2.	Какая из перечисленных функций не может быть конструктором?	1. void String (); 2. String (); 3. String (String & s); 4. String (const int a).
3.	Отметьте правильное утверждение для абстрактного класса для языка C++:	1. Класс, у которого все методы чисто виртуальные, называется абстрактным; 2. Абстрактный базовый класс навязывает определенный интерфейс всем производным из него классам; 3. Невозможно создать объект абстрактного класса; 4. В абстрактном классе не описываются методы вообще.
4.	Если в программе на языке C++ в производном классе переопределена операция new то...	1. Все объекты этого класса и все объекты классов, выведенных из него, будут использовать эту операцию независимо от зоны видимости, в которой она переопределена; 2. Производные от этого класса могут использовать глобальную операцию применив операцию базовый класс :: new; 3. Операцию new нельзя переопределить; 4. В любом случае эта операция будет доступна только в пределах класса-потомка.
5.	Какой из перечисленных методов может быть конструктором для класса String в языке C++?	1. String * String (); 2. void String (); 3. String (String & s); 4. const String (int a).
6.	Какая функция, не будучи компонентом класса, имеет доступ к его защищенным и внутренним компонентам?	1. Шаблонная; 2. Полиморфная; 3. Дружеская; 4. Статическая.

7.	Укажите правильное объявление виртуального метода, который принимает одно целочисленное значение и возвращает void:	<ol style="list-style-type: none"> <li>1. <code>virtual void SomeFunction (int x);</code></li> <li>2. <code>void SomeFunction (int x) virtual;</code></li> <li>3. <code>virtual SomeFunction (int x);</code></li> <li>4. <code>virtual void SomeFunction (int * x).</code></li> </ol>
8.	Принцип объектно-ориентированного программирования, заключаемый в объединении атрибутов и методов объекта с целью обеспечения сохранности данных, называется...	<ol style="list-style-type: none"> <li>1. Наследование;</li> <li>2. Сочетание;</li> <li>3. Инициализация;</li> <li>4. Инкапсуляция.</li> </ol>
9.	В программе описано абстрактный класс A и производный от этого класса класс A1. Какой из записей заведомо неверный?	<ol style="list-style-type: none"> <li>1. <code>A * a = new A;</code></li> <li>2. <code>A1 a1;</code></li> <li>3. <code>A1 a1; A &amp; a = a1;</code></li> <li>4. <code>A1 a1; A1 a2.</code></li> </ol>
10.	Выберите верное утверждение о деструкторе класса в C++:	<ol style="list-style-type: none"> <li>1. Деструктор принимает в качестве параметра адрес того объекта, который нужно уничтожить;</li> <li>2. Деструктор принимает в качестве параметра указатель <code>this</code>;</li> <li>3. Деструктор не содержит параметров;</li> <li>4. Деструктор принимает в качестве параметра тот объект, который нужно уничтожить.</li> </ol>
11.	Какой из вариантов записи абстрактного класса в C++ является правильным?	<ol style="list-style-type: none"> <li>1. <code>abstract class A { virtual int f () = 0;};</code></li> <li>2. <code>class A { virtual int f () = 0;};</code></li> <li>3. <code>class A { virtual int f () = 0;} abstract;</code></li> <li>4. <code>class A { virtual int f ();}.</code></li> </ol>
12.	Программа – это...	<ol style="list-style-type: none"> <li>1. Система правил, описывающая последовательность действий, которые необходимо выполнить для решения задачи;</li> <li>2. Указание на выполнение действий из заданного набора;</li> <li>3. Область внешней памяти для хранения текстовых, числовых данных и другой информации;</li> <li>4. Последовательность команд, реализующая алгоритм решения задачи.</li> </ol>
13.	Имя переменной – это...	<ol style="list-style-type: none"> <li>1. Любая последовательность любых символов;</li> <li>2. Последовательность латинских букв, цифр, специальных знаков (кроме пробела);</li> <li>3. Последовательность, которая всегда должна начинаться с латинской буквы;</li> <li>4. Последовательность русских, латинских букв, начинающихся с латинской буквы и из специальных знаков, допускающих знак подчеркивания.</li> </ol>
14.	Какой тип переменной используется в коде: <code>int a = 5?</code>	<ol style="list-style-type: none"> <li>1. Знаковое 8-бит целое;</li> <li>2. Знаковое 64-бит целое;</li> <li>3. Знаковое 32-бит целое;</li> <li>4. 1 байт*.</li> </ol>

15.	Что делает оператор «%»?	<ol style="list-style-type: none"> <li>1. Возвращает процент от суммы;</li> <li>2. Возвращает остаток от деления;</li> <li>3. Возвращает тригонометрическую функцию;</li> <li>4. Ничего из выше перечисленного.</li> </ol>
16.	Что сделает программа выполнив следующий код: Console.WriteLine(«Hello, World!»)?	<ol style="list-style-type: none"> <li>1. Напишет на новой строке Hello, World!;</li> <li>2. Напишет Hello, World!;</li> <li>3. Удалит все значения с Hello, World!;</li> <li>4. Вырежет слово Hello, World! из всего текста.</li> </ol>
17.	Как найти квадратный корень из числа x?	<ol style="list-style-type: none"> <li>1. Sqrt(x);</li> <li>2. Summ.Koren(x);</li> <li>3. Arifmetic.sqrt(x);</li> <li>4. Math.Sqrt(x).</li> </ol>
18.	Обозначения оператора «НЕ»:	<ol style="list-style-type: none"> <li>1. Not;</li> <li>2. No;</li> <li>3. !;</li> <li>4. !=.</li> </ol>
19.	Обозначение оператора «ИЛИ»:	<ol style="list-style-type: none"> <li>1. !;</li> <li>2. !=;</li> <li>3.   ;</li> <li>4. Or.</li> </ol>
20.	Для чего нужны условные операторы?	<ol style="list-style-type: none"> <li>1. Чтобы устанавливать условия пользователю;</li> <li>2. Для ветвления программы;</li> <li>3. Для оптимизации программы;</li> <li>4. Используются по умолчанию.</li> </ol>

### Вариант №3

№ п/п	Вопрос	Варианты ответа
1.	Что такое массив?	<ol style="list-style-type: none"> <li>1. Набор однотипных данных, которые располагаются в памяти последовательно друг за другом;</li> <li>2. Набор текстовых значений в формате Unicode, которые расположены в случайном порядке;</li> <li>3. Набор данных типа int (32-бит целое);</li> <li>4. Переменная.</li> </ol>
2.	Какими бывают массивы?	<ol style="list-style-type: none"> <li>1. Разнообразные;</li> <li>2. Сложные и простые;</li> <li>3. Одномерные и многомерные;</li> <li>4. Резиновые и статичные.</li> </ol>
3.	Для чего нужны циклы?	<ol style="list-style-type: none"> <li>1. Циклы нужны для многократного запуска программы;</li> <li>2. Циклы нужны для многократного выполнения кода;</li> <li>3. Циклы нужны для многократного размещения данных;</li> <li>4. Циклы нужны чтобы выполнить код без ошибок.</li> </ol>

4.	Какой оператор возвращает значение из метода?	<ol style="list-style-type: none"> <li>1. <code>veni</code>;</li> <li>2. <code>return</code>;</li> <li>3. <code>out</code>;</li> <li>4. <code>end</code>.</li> </ol>
5.	Что такое константа?	<ol style="list-style-type: none"> <li>1. Переменная типа <code>string</code>;</li> <li>2. Переменная которая может быть изменена в любое время;</li> <li>3. Глобальная переменная;</li> <li>4. Переменная значение которой нельзя изменить.</li> </ol>
6.	Класс – это:	<ol style="list-style-type: none"> <li>1. Любой тип данных, определяемый пользователем;</li> <li>2. Тип данных, определяемый пользователем и сочетающий в себе данные и функции их обработки;</li> <li>3. Структура, для которой в программе имеются функции работы с нею;</li> <li>4. Любой тип данных, определяемый программой.</li> </ol>
7.	Членами класса могут быть:	<ol style="list-style-type: none"> <li>1. Как переменные, так и функции, могут быть объявлены как <code>private</code> и как <code>public</code>;</li> <li>2. Только переменные, объявленные как <code>private</code>;</li> <li>3. Только функции, объявленные как <code>private</code>;</li> <li>4. Только переменные и функции, объявленные как <code>private</code>.</li> </ol>
8.	Какие диаграммы служат основой для генерации кода на целевом языке программирования?	<ol style="list-style-type: none"> <li>1. Диаграммы вариантов использования;</li> <li>2. Диаграммы последовательности;</li> <li>3. Диаграммы классов;</li> <li>4. Диаграммы состояний.</li> </ol>
9.	Динамика конкретного класса может быть выражена с помощью ...	<ol style="list-style-type: none"> <li>1. Диаграммы вариантов использования;</li> <li>2. Диаграммы последовательности;</li> <li>3. Диаграммы классов;</li> <li>4. Диаграммы состояний.</li> </ol>
10.	Какая функция, не будучи компонентом класса, имеет доступ к его защищенным и внутренним компонентам?	<ol style="list-style-type: none"> <li>1. Шаблонная;</li> <li>2. Полиморфная;</li> <li>3. Дружеская;</li> <li>4. Статическая.</li> </ol>
11.	Принцип объектно-ориентированного программирования, заключающийся в объединении атрибутов и методов объекта с целью обеспечения сохранности данных, называется:	<ol style="list-style-type: none"> <li>1. Наследование;</li> <li>2. Сочетание;</li> <li>3. Инициализация;</li> <li>4. Инкапсуляция.</li> </ol>
12.	Выберите верное утверждение о деструкторе класса в C++:	<ol style="list-style-type: none"> <li>1. Деструктор принимает в качестве параметра адрес того объекта, который нужно уничтожить;</li> <li>2. Деструктор принимает в качестве параметра указатель <code>this</code>;</li> <li>3. Деструктор не содержит параметров;</li> <li>4. Деструктор принимает в качестве параметра тот объект, который нужно уничтожить.</li> </ol>

13.	Укажите тип данных, который определяет служебное слово <code>struct</code> :	<ol style="list-style-type: none"> <li>1. Массив данных с различной структурой;</li> <li>2. Тип функций, которые могут иметь различную структуру параметров;</li> <li>3. Тип данных, которые могут менять свою структуру;</li> <li>4. Составной объект, к которому могут входить элементы различных типов.</li> </ol>
14.	Укажите типы доступа возможные для элементов структур:	<ol style="list-style-type: none"> <li>1. Любые;</li> <li>2. Только закрытые;</li> <li>3. Только открытые;</li> <li>4. Только защищенные.</li> </ol>
15.	Определите, для чего предназначены конструкторы класса:	<ol style="list-style-type: none"> <li>1. Для конструирования класса;</li> <li>2. Для инициализации объектов класса;</li> <li>3. Для выделения памяти объектам класса;</li> <li>4. Для объявления объектов класса.</li> </ol>
16.	Определите, что такое производный класс:	<ol style="list-style-type: none"> <li>1. Класс, построенный на основе другого класса;</li> <li>2. Класс, на основе которого построен другой класс;</li> <li>3. Класс, на основе которого построено ни одного класса;</li> <li>4. Класс, использующий методы.</li> </ol>
17.	Укажите, каких спецификаторов доступа не существует:	<ol style="list-style-type: none"> <li>1. <code>public</code>;</li> <li>2. <code>protected</code>;</li> <li>3. <code>private</code>;</li> <li>4. <code>open</code>.</li> </ol>
18.	Определите, что такое множественное наследование:	<ol style="list-style-type: none"> <li>1. Наследование нескольких производных классов от одного базового;</li> <li>2. Наследования производного класса от нескольких базовых;</li> <li>3. Наследование производного класса с спецификатором <code>manu</code>;</li> <li>4. Наследование производного класса от класса, который является производным от другого класса.</li> </ol>
19.	Имеет ли доступ производный класс к закрытым элементам своего базового класса?	<ol style="list-style-type: none"> <li>1. Имеет;</li> <li>2. Нет;</li> <li>3. Имеет при открытом наследовании;</li> <li>4. Имеет при отсутствии спецификатора наследования.</li> </ol>
20.	В сложных выражениях последовательность выполнения операций определяется...	<ol style="list-style-type: none"> <li>1. Только приоритетом операций;</li> <li>2. Только скобками;</li> <li>3. Скобками, приоритетом операций, а при одинаковом приоритете ассоциативностью операций;</li> <li>4. Только ассоциативностью операций.</li> </ol>

### 6.3. Описание показателей и критериев контроля успеваемости, описание шкал оценивания

#### 6.3.1. Критерии оценок промежуточной аттестации (экзамен)

Оценка			
«2» (неудовлетворительно)	Пороговый уровень освоения	Углубленный уровень освоения	Продвинутый уровень освоения
	«3» (удовлетворительно)	«4» (хорошо)	«5» (отлично)
Студент не знает значительной части материала, допускает существенные ошибки в ответах на вопросы	Студент поверхностно знает материал основных разделов и тем учебной дисциплины, допускает неточности в ответе на вопрос	Студент хорошо знает материал, грамотно и по существу излагает его, допуская некоторые неточности в ответе на вопрос.	Студент в полном объеме знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос
Не умеет находить решения большинства предусмотренных программой обучения заданий	Иногда находит решения, предусмотренные программой обучения заданий	Уверенно находит решения, предусмотренные программой обучения заданий	Безошибочно находит решения, предусмотренные программой обучения заданий
Большинство предусмотренных программой обучения заданий не выполнено	Предусмотренные программой обучения задания выполнены удовлетворительно	Предусмотренные программой обучения задания успешно выполнены	Предусмотренные программой обучения задания успешно выполнены

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 7.1. Рекомендуемая литература

#### 7.1.1. Основная литература

1. Павловская Т.А. С/С++Программирование на языке высокого уровня. – СПб.: Питер. 2003. – 461 с.
2. Прата С. Язык программирования С++. Лекции и упражнения – Москва.: Вильямс. 2015. – 928 с.
3. Павлов Л.А., Первова Н.В. Структуры и алгоритмы обработки данных. – СПб.: Лань. 2020. – 256 с.
4. Игошин В.И. Теория алгоритмов. – Москва.: Инфра-М. 2016. – 318 с.
5. Стивенс Р. Алгоритмы. Теория и практическое применение. – Москва.: Эксмо. 2016. – 544 с.
6. Солтис М. Введение в анализ алгоритмов. – Москва.: ДМК. 2019. – 278 с.
7. Клейнберг Д., Тардос Э. Алгоритмы: разработка и применение. – СПб.: Питер. 2016. – 800 с.

#### 7.1.2. Дополнительная литература

8. Ершов И.А. Реализация программы «Цифровая экономика» в системе обеспечения единства измерений // Автоматика и программная инженерия – 2018 № 1. С. 105-109.
9. Филиппов В.П., Музалевский А.В. Потенциальные угрозы информационной безопасности цифровой экономики в системе метрологического обеспечения // Труды КГНЦ – 2020 № 1. С. 209-218.

10. Ковригин Е.А., Васильев В.А. Пути развития СМК в условиях цифровизации // Компетентность. – 2020. № 6. С. 12-17.

11. Горячева М.Ю. Использование автоматизированного учета средств измерения для облегчения метрологического обеспечения производства // Международный журнал гуманитарных и естественных наук – 2017. № 12. С. 135-138.

12. Горячева М.Ю. Использование автоматизированного учета средств измерения для облегчения метрологического обеспечения производства // Международный журнал гуманитарных и естественных наук – 2017. № 12. С. 135-138.

### **7.1.3. Учебно-методическое обеспечение**

13. Ершов И.А. Реализация программы «Цифровая экономика» в системе обеспечения единства измерений // Автоматика и программная инженерия - 2018 № 1. С. 105-109.

## **7.2. Базы данных, электронно-библиотечные системы, информационно-справочные и поисковые системы**

1. <https://www.geeksforgeeks.org/must-do-coding-questions-for-companies-like-amazon-microsoft-adobe/>

2. <https://www.hackerearth.com/practice/codemonk/>

3. <https://www.interviewbit.com/courses/programming/>

4. <https://www.spoj.com/problems/classical/>

5. <https://app.codility.com/programmers/>

6. <https://welearncode.com/>

7. <http://codeforces.com/blog/entry/13529>

8. <https://discuss.codechef.com/t/data-structures-and-algorithms/6599>

9. <https://github.com/ZoranPandovski/al-go-rithms>

10. <https://www.hackerrank.com/domains/cpp>

11. <https://www.programiz.com/cpp-programming>

12. <https://hackr.io/tutorials/learn-c-plus-plus>

## **8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Специализированные аудитории, используемые при проведении занятий лекционного типа, практических занятий, оснащены мультимедийными проекторами и комплектом аппаратуры, позволяющей демонстрировать текстовые и графические материалы.

### **8.1. Материально-техническое оснащение аудиторий:**

#### **Аудитории для проведения лекционных занятий:**

*33 посадочных места*

Оснащенность: Стол аудиторный – 18 шт., стул аудиторный – 32 шт., доска настенная – 1 шт., стул преподавателя – 1 шт., Мультимедийный комплекс – 1 шт.

*71 посадочное место*

Оснащенность: Стол аудиторный – 31 шт., стул аудиторный – 70 шт., стул преподавателя – 1 шт., Мультимедийный комплекс – 1 шт.



### **Аудитории для проведения практических занятий:**

#### *19 посадочных мест*

Оснащенность: Стол аудиторный – 11 шт., стул аудиторный – 18 шт., доска настенная – 1 шт., кресло преподавателя – 1 шт., компьютеры – 19 шт. с возможность подключения к сети «Интернет», лазерный принтер – 1шт, шкаф – 4 шт.

#### *25 посадочных мест*

Оснащенность: Стол аудиторный – 14 шт., стул аудиторный – 24 шт., доска мобильная – 1 шт., кресло преподавателя – 1 шт., компьютеры – 25 шт. с возможность подключения к сети «Интернет», принтер – 1шт.

### **8.2. Помещения для самостоятельной работы :**

1. Оснащенность помещения для самостоятельной работы: 12 посадочных мест. Стул – 12 шт., стол – 6 шт., шкаф – 8 шт., АРМ учебное ПК (монитор + системный блок) – 12 шт. Доступ к сети «Интернет», в электронную информационно-образовательную среду Университета, принтер – 1 шт.

Перечень лицензионного программного обеспечения: Microsoft Windows 7 Professional: ГК № 1464-12/10 от 15.12.10 «На поставку компьютерного оборудования» ГК № 959-09/10 от 22.09.10 «На поставку компьютерной техники» (обслуживание до 2025 года) ГК № 447-06/11 от 06.06.11 «На поставку оборудования» (обслуживание до 2025 года) ГК № 984-12/11 от 14.12.11 «На поставку оборудования" (обслуживание до 2025 года) Договор № 1105-12/11 от 28.12.2011 «На поставку компьютерного оборудования» (обслуживание до 2025 года), Договор № 1106-12/11 от 28.12.2011 «На поставку компьютерного оборудования» (обслуживание до 2025 года) ГК № 671-08/12 от 20.08.2012 «На поставку продукции» (обслуживание до 2025 года), Microsoft Open License 60799400 от 20.08.2012, Microsoft Open License 48358058 от 11.04.2011, Microsoft Open License 49487710 от 20.12.2011, Microsoft Open License 49379550 от 29.11.2011 (обслуживание до 2025 года),

Microsoft Office 2010 Standard: Microsoft Open License 60799400 от 20.08.2012 (обслуживание до 2025 года), Microsoft Open License 60853086 от 31.08.2012 (обслуживание до 2025 года)

Kaspersky antivirus 6.0.4.142

### **8.3. Помещения для хранения и профилактического обслуживания оборудования:**

1. Центр новых информационных технологий и средств обучения:

Оснащенность: персональный компьютер – 2 шт. (доступ к сети «Интернет»), монитор – 4 шт., сетевой накопитель – 1 шт., источник бесперебойного питания – 2 шт., телевизор плазменный Panasonic – 1 шт., точка Wi-Fi – 1 шт., паяльная станция – 2 шт., дрель – 5 шт., перфоратор – 3 шт., набор инструмента – 4 шт., тестер компьютерной сети – 3 шт., баллон со сжатым газом – 1 шт., паста теплопроводная – 1 шт., пылесос – 1 шт., радиостанция – 2 шт., стол – 4 шт., тумба на колесиках – 1 шт., подставка на колесиках – 1 шт., шкаф – 5 шт., кресло – 2 шт., лестница Alve – 1 шт.

Перечень лицензионного программного обеспечения: Microsoft Windows 7 Professional (Лицензионное соглашение Microsoft Open License 60799400 от 20.08.2012)

Microsoft Office 2010 Professional Plus (Лицензионное соглашение Microsoft Open License 60799400 от 20.08.2012)

Антивирусное программное обеспечение Kaspersky Endpoint Security (Договор № Д810(223)-12/17 от 11.12.17)

2. Центр новых информационных технологий и средств обучения:

Оснащенность: стол – 5 шт., стул – 2 шт., кресло – 2 шт., шкаф – 2 шт., персональный компьютер – 2 шт. (доступ к сети «Интернет»), монитор – 2 шт., МФУ – 1 шт., тестер компьютерной сети – 1 шт., баллон со сжатым газом – 1 шт., шуруповерт – 1 шт.

Перечень лицензионного программного обеспечения: Microsoft Windows 7 Professional (Лицензионное соглашение Microsoft Open License 60799400 от 20.08.2012)

Microsoft Office 2007 Professional Plus (Лицензионное соглашение Microsoft Open License 46431107 от 22.01.2010)

Антивирусное программное обеспечение Kaspersky Endpoint Security (Договор № Д810(223)-12/17 от 11.12.17)

3. Центр новых информационных технологий и средств обучения:

Оснащенность: стол – 2 шт., стулья – 4 шт., кресло – 1 шт., шкаф – 2 шт., персональный компьютер – 1 шт. (доступ к сети «Интернет»), веб-камера Logitech HD C510 – 1 шт., колонки Logitech – 1 шт., тестер компьютерной сети – 1 шт., дрель – 1 шт., телефон – 1 шт., набор ручных инструментов – 1 шт.

Перечень лицензионного программного обеспечения: Microsoft Windows 7 Professional (Лицензионное соглашение Microsoft Open License 48358058 от 11.04.2011)

Microsoft Office 2007 Professional Plus (Лицензионное соглашение Microsoft Open License 46431107 от 22.01.2010)

Антивирусное программное обеспечение Kaspersky Endpoint Security (Договор № Д810(223)-12/17 от 11.12.17)

#### **8.4. Лицензионное программное обеспечение:**

1. Microsoft Office Std 2010 RUS (Контракт № 0372100009514000092-0003177-01 от 02.09.2014)

2. Microsoft Office Std 2013 RUS OLP NL Acsmc (Контракт № 0372100009515000100-0003177-01 от 26.06.2015 года)

3. Операционная система Microsoft Windows Pro 7 PRO RUS (Контракт № 0372100009514000092-0003177-01 от 02.09.2014)

4. Операционная система Лицензия Windows 8 Pro 32-bit/64-bit (Контракт № 0372100009515000100-0003177-01 от 26.06.2016 года)

5. Антивирусное программное обеспечение ESET NOD32 Smart Security Business Edition newsale (Договор № 0372100009513000040-0003177-02 от 05.11.2017 года, Контракт № 0372100009514000092-0003177-01 от 02.09.2014, Контракт № 0372100009515000100-0003177-01 от 26.06.2017 года).