**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Язык программирования С++

C++ Programming Language

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 2

Регистрационный номер рабочей программы: 002240

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Целью дисциплины «Язык программирования С++» является обучение возможностям языка программирования С++ и основным приемам программирования на этом языке. Задачей дисциплины является также приобретение обучающимися базовых практических навыков программирования на С++, знакомство обучающихся с основными типичными проблемами, возникающими при программировании на языке С++ и способами их решения.

Отдельные параметры односеместрового курса могут варьироваться по степени сложности в зависимости от начальной подготовки обучающихся.

По окончании обучения обучающиеся должны знать содержание дисциплины «Язык программирования С++», иметь достаточно полное представление об основных возможностях и приемах программирования языка С++ и уметь решать типовые задачи.

**1.2. Требования подготовленности обучающегося к освоению содержания учебных занятий (пререквизиты)**

Программа дисциплины рассчитана на обучающихся 3–ого курса. Максимальная эффективность программы будет обеспечена при следующем условии: обучающийся владеет базовыми навыками программирования и основами объектно-ориентированного программирования.

**1.3. Перечень результатов обучения (learning outcomes)**

* знание основных возможностей языка С++, умение применять их для решения практических задач;
* владение основными приемами и методами программирования на С++;
* знание типичных трудностей и ошибок при программировании на С++ и умение их избегать;
* знание основных направлений развития языка.

Дисциплина способствует развитию следующих компетенций:

* ОПК-1 — способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности;
* ОПК-3 — способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения;
* ОПК-4 — способен участвовать в разработке технической документации программных продуктов и программных комплексов;
* ОПК-5 — способен инсталлировать и сопровождать программное обеспечение для информационных систем и баз данных, в том числе отечественного производства;
* ПКА-1 — способен демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий;
* ПКА-2 — способен учитывать знания проблем и тенденций развития рынка ПО в профессиональной деятельности;
* ПКП-1 — способность проводить под научным руководством исследование на основе существующих методов в конкретной области профессиональной деятельности;
* ПКП-4 — способен применять современные информационные технологии при проектировании, реализации, оценке качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях;
* ПКП-5 — способен использовать основные методы и средства автоматизации проектирования, реализации, испытаний и оценки качества при создании конкурентоспособного программного продукта и программных комплексов, а также способен использовать методы и средства автоматизации, связанные с сопровождением, администрированием и модернизацией программных продуктов и программных комплексов;
* ПКП-6 — способен использовать знания направлений развития компьютеров с традиционной (нетрадиционной) архитектурой; современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ; тенденции развития функций и архитектур проблемно-ориентированных программных систем и комплексов в профессиональной деятельности;
* УКБ-3 — способен понимать сущность и значение информации в развитии общества, использовать основные методы получения и работы с информацией с учетом современных технологий цифровой экономики и информационной безопасности.

**1.4. Перечень и объём активных и интерактивных форм учебных занятий**

Активные и интерактивные занятия: семинары (28 ак. часов).

**Раздел 2. Организация, структура и содержание учебных занятий**

**2.1. Организация учебных занятий**

**2.1.1 Основной курс**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Трудоёмкость, объёмы учебной работы и наполняемость групп обучающихся | | | | | | | | | | | | | | | | | | |
| Код модуля в составе дисциплины,  практики и т.п. | Контактная работа обучающихся с преподавателем | | | | | | | | | | | | Самостоятельная работа | | | | Объём активных и интерактивных  форм учебных занятий | Трудоёмкость |
| лекции | семинары | консультации | практические  занятия | лабораторные работы | контрольные работы | коллоквиумы | текущий контроль | промежуточная  аттестация | итоговая аттестация | под руководством преподавателя | в присутствии  преподавателя | сам. раб. с использованием  методических материалов | текущий контроль (сам.раб.) | промежуточная аттестация (сам.раб.) | итоговая аттестация  (сам.раб.) |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | | | | | | | | | | | | | |
| Форма обучения: очная | | | | | | | | | | | | | | | | | | |
| Семестр 6 |  | 30 |  |  |  |  |  |  | 2 |  |  |  | 32 |  | 8 |  | 28 | 2 |
|  |  | 10-25 |  |  |  |  |  |  | 10-25 |  |  |  | 1-1 |  | 1-1 |  |  |  |
| ИТОГО |  | 30 |  |  |  |  |  |  | 2 |  |  |  | 32 |  | 8 |  |  | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Виды, формы и сроки текущего контроля успеваемости и промежуточной аттестации | | | | | | |
| Код модуля в составе дисциплины, практики и т.п. | Формы текущего контроля успеваемости | | Виды промежуточной аттестации | | Виды итоговой аттестации  (только для программ итоговой аттестации и дополнительных образовательных программ) | |
| Формы | Сроки | Виды | Сроки | Виды | Сроки |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | |
| Форма обучения: очная | | | | | | |
| Семестр 6 |  |  | зачёт, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

1. Основные типы и операции над ними. Основные операторы: семинар 2 ч.

Типы для целых и вещественных чисел. Типичные ошибки при работе с числами. Операции. Порядок вычисления выражений. Особенности вычисления && и ||.

1. Указатели, динамическая память, ссылки. Функции: семинар 3 ч.

Операторы new и delete. Арифметические операции с указателями. Работа с массивами с помощью указателей. Описание функций. inline функции. Параметры по умолчанию. Перегрузка функций. Ссылки. Примеры их использования – в качестве переменной, в качестве параметра, в качестве результата функции.

1. Классы. Наследование классов. Виртуальные функции: семинар 3 ч.

Основные возможности для описания классов. Списки инициализации. Конструктор по умолчанию. Деструктор. Наследование. Виртуальные функции. Виртуальные деструкторы. Реализация виртуальных функций, таблицы виртуальных методов (vtable). Чисто виртуальные функции. Множественное наследование. private и protected наследование. Принцип подстановочности Лисков.

1. Исключения и обработка ошибок: семинар 2 ч.

Конструкции throw и try / catch. Конструкции catch (...) и throw; Макро assert. Освобождение ресурсов при обработке исключений. Принцип RAII.

1. Константы в С++: семинар 1 ч.

Определение констант. Перечисления. Константы и указатели. const при описании полей класса. const при описании методов класса. Описатель mutable.

1. Дополнительные возможности описания классов: семинар 2 ч.

Описатель friend. Перегрузка операторов. Определение преобразований типа с помощью конструктора и с помощью перегрузки оператора преобразования. Описатель explicit. static поля и методы.

1. Особенности реализации классов, использующих динамическую память: семинар 2 ч.

Необходимость переопределения оператора =. Конструктор копирования. Запрещение оператора = и конструктора копирования. «Правило 3х функций». Разделяемые данные. Понятие счетчика указателей.

1. Стандартная библиотека шаблонов: семинар 4 ч.

Шаблоны list, vector, deque. Их сравнение и области их применения. Итераторы, работа с контейнерами с помощью итераторов. const\_iterator, value\_type. Шаблоны set, map, multiset, multimap. Их сравнение и области их применения. Основные операции. Пример: сортировка по вычисляемому свойству c помощью multimap. Принципы организации библиотеки алгоритмов. Функции, как параметры алгоритмов. Примеры: функция sort, accumulate, find, find\_if.

1. Указатели на функции и объекты - функции: семинар 2 ч.

Указатели на функции. Функции, как параметры других функций. Объекты - функции. Использование объектов – функций в алгоритмах STL.

1. Шаблоны: семинар 5 ч.

Шаблоны функций. Шаблоны классов. Специализация шаблонов. Параметры шаблона по умолчанию.

Частичная специализация. Описатель typename. Понятие класса свойств (traits).

1. Преобразования типов, union, RTTI: семинар 2 ч.

Операторы преобразования: static\_cast, reinterpret\_cast, const\_cast, dynamic\_cast. Динамическая идентификация типа (RTTI). Оператор typeid.

1. Программы из нескольких исходных файлов. Пространства имен. Препроцессор: семинар 4 ч.

Создание проектов из нескольких исходных файлов. Описатель extern. Описатель static для глобальных переменных и функций. Пространства имен и их использование. Безымянные пространства имен. Основные операторы препроцессора. Операторы # и ##

1. Обзор возможностей стандарта C++11, стандарта С++17 и библиотеки Boost: семинар 4 ч.

«Умные указатели». Автоматическое определение типа. Лямбда выражения. Понятие rvalue. Обзор библиотеки boost.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Методические материалы включают в себя следующие типы материалов: методические указания для обучающихся, презентации семинаров.

**3.1.2 Методическое обеспечение самостоятельной работы**

Методические материалы включают в себя следующие типы материалов: Интернет-ресурсы, задачи для самостоятельного решения и примеры решений, фонд контрольных заданий для самоконтроля.

**3.1.3 Методика проведения текущего контроля успеваемости и промежуточной аттестации и критерии оценивания**

Промежуточная аттестация проводится в виде письменной работы. Работа состоит из 10 заданий, каждое оценивается по шкале от 0 до 10 баллов. На решение заданий отводится с общей сложности два академических часа, разрешается пользоваться источниками.

Итоговый процент освоения дисциплины вычисляется по формуле . Таким образом, обучающийся, успешно решивший менее 50% заданий, получает 0, обучающийся, решивший 8 задач, получает 60, обучающийся, успешно справившийся со всеми задачами, получает 100. Далее применяется шкала оценивания ECTS:

|  |  |  |
| --- | --- | --- |
| Процент освоения дисциплины | Оценка ECTS | Оценка при проведении зачёта |
| от 90% до 100% | A | зачтено |
| от 80% до 89% | B |
| от 70% до 79% | C |
| от 60% до 69% | D |
| от 50% до 59% | E |
| менее 50% | F | не зачтено |

**3.1.4 Методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации (контрольно-измерительные материалы, оценочные средства)**

Для проведения промежуточной аттестации проводится заключительная контрольная работа. Контрольная работа включает в себя задачи по следующим темам:

1. Выражения и порядок их вычисления. Операции ?: и , . Разница между i++ и ++i. Особенности вычисления && и ||.
2. Указатели, работа с динамической памятью. Арифметические операции с указателями. Работа с массивами с помощью указателей. Массивы указателей.
3. Функции. В частности, функции для обработки массивов.
4. Ссылки: как параметры, как результат функции.
5. Работа со строками.
6. Классы.
7. Наследование, виртуальные функции, чисто виртуальные функции.
8. Контейнеры + наследование.
9. Исключения.
10. Списки и деревья.
11. Перегрузка операторов. Определение пользовательских операторов преобразования. friend.
12. Почему оператор + лучше описывать не как метод, а как обычную функцию? Переопределение операторов типа = (почему принято писать return \*this;). = default; и = delete;
13. Особенности реализации классы, использующие динамическую память (Скорее всего не примере string, который мы сами определяли). Правило трех функций.
14. Разделяемые данные (счетчик указателей).
15. Ассоциативные контейнеры: map, multimap.
16. Программирование шаблонов для работы к контейнерами STL. value\_type. decltype.
17. Функции, как параметры других функций. Использование лямбда выражений. Список захвата.
18. Функция sort. Лямбда-выражения в sort.
19. Фунцкции find\_if, all\_of, any\_of.
20. STL: vector, list.
21. Шаблоны функций и классов.
22. Специализация шаблонов функций и классов. Частичная специализация. Понятие trait.
23. Преобразования типов. RTTI (dynamic\_cast и typeid).
24. Программы из нескольких модулей.
25. Умные указатели и работа с ними. shared\_ptr, unique\_ptr, std::move
26. Понятие RAII
27. Новые возможности стандарта C++17

**3.1.5 Методические материалы для оценки обучающимися содержания и качества учебного процесса**

Для оценки обучающимися содержания и качества учебного процесса применяется анкетирование в соответствии с методикой и графиком, утвержденными в установленном порядке.

**3.2. Кадровое обеспечение**

**3.2.1 Образование и (или) квалификация штатных преподавателей и иных лиц, допущенных к проведению учебных занятий**

К преподаванию дисциплины могут быть допущены преподаватели, имеющие диплом о высшем образовании по соответствующему направлению.

**3.2.2 Обеспечение учебно-вспомогательным и (или) иным персоналом**

Специальных требований нет

**3.3. Материально-техническое обеспечение**

**3.3.1 Характеристики аудиторий (помещений, мест) для проведения занятий**

В аудиториях, где проводятся занятия, необходимо наличие досок.

**3.3.2 Характеристики аудиторного оборудования, в том числе неспециализированного компьютерного оборудования и программного обеспечения общего пользования**

В аудиториях, где проводятся занятия, желательно наличие проектора для показа презентаций.  
**3.3.3 Характеристики специализированного оборудования**

Специальных требований нет

**3.3.4 Характеристики специализированного программного обеспечения**

Специальных требований нет

**3.3.5 Перечень и объёмы требуемых расходных материалов**

Расходные материалы не требуются

**3.4. Информационное обеспечение**

**3.4.1 Список обязательной литературы**

1. Страуструп Б. Язык программирования С++. «Бином», 2008.

2. Саттер Г., Александреску А. Стандарты программирования на C++. М., «Вильямс», 2005.

3. Лаптев В. C++. Экспресс курс. «БХВ-Петербург», 2004.

**3.4.2 Список дополнительной литературы**

1. Саттер Г. Решение сложных задач на С++. М., «Вильямс», 2008.

2. Саттер Г. Новые сложные задачи на С++. М., «Вильямс», 2005.

3. Липпман С., Лажойе Ж., Му Б. Язык программирования С++. Вводный курс. М., «Вильямс», 2007.

**3.4.3 Перечень иных информационных источников**

Ресурсы сети Интернет.

**Раздел 4. Разработчики программы**

Салищев Сергей Игоревич, к. ф.-м. н, старший преподаватель кафедры информатики, s.salischev@spbu.ru

По материалам Симуни Михаила Лазаревича.