

## **Аннотация к рабочей программе дисциплины Б1.В.ОД.12 «Современные языки программирования»**

### *Цель изучения дисциплины:*

освоение студентами языка C++ и на его основе овладение основными приёмами и методами программирования и алгоритмизации;

приобретение навыков самостоятельного изучения отдельных тем дисциплины и решения типовых задач;

приобретение навыков работы в современных интегрированных системах программирования;

приобретение навыков разработки программных комплексов;

усвоение полученных знаний студентами, а также формирование у них мотивации к самообразованию за счет активизации самостоятельной познавательной деятельности.

### *Место дисциплины в структуре ОП*

Дисциплина «Современные языки программирования» является обязательной дисциплиной вариативной части по направлению 09.03.01 «Информатика и вычислительная техника».

### *Содержание дисциплины:*

Основы алгоритмизации. Модели решения функциональных и вычислительных задач. Основные этапы решения задач на ЭВМ; алгоритмы и их свойства. Структура программы. Алфавит языка. Имена. Целые типы. Плавающие типы данных. Директивы препроцессора. Именованные константы. Операции C++. Операции присваивания, сравнения, арифметические, логические и т.д. Операторы C++. Условные операторы. Ветвление и циклы. Переключатели. Указатели и массивы. Адресная арифметика. Многомерные массивы и символьные строки. Динамическое распределение памяти. Функции. Передача аргументов по значению. Объявления и определения. Область существования и видимости имен. Классы памяти. Объявления объектов и типов. Правила преобразования стандартных типов. Ссылки. Функции. Передача аргументов по ссылке. Аргументы по умолчанию. Передача указателя на функцию в качестве аргумента. Классы. Понятие абстрактных типов данных. Скрытые, общие и защищенные данные. Операции ".", "->". Область видимости класс. Функции-члены классов. Инлайн-функции. Инициализация данных. Конструкторы и деструкторы, их перегрузка. Порядок выполнения конструкторов и деструкторов. Статические члены класса. Указатель this. Статические функции-члены. Указатели на члены класса. Дружественные функции и дружественные классы. Конструктор и операция NEW. Наследование. Построение производного класса. Защищенные члены. Раннее и позднее (динамическое) связывание. Виртуальные функции. Чисто абстрактные классы. Переопределение стандартных операций. Особенности переопределения операций new, delete, =, [], (), ->. Преобразования абстрактных типов. Оператор-функция преобразования типов. Способы эффективного хранения и обработки данных. Основные структуры данных и их обработка с точки зрения объектно-ориентированного программирования. Списки. Двухнаправленные и кольцевые

списки. Их построение и реализация. Очереди и стеки. Их построение и реализация в виде массива и списка. Деревья. Двоичное дерево поиска. Их построение и реализация. Таблицы. Их применение в поисковых алгоритмах построения баз данных. Оценка качества программных продуктов. Методы защиты программ и данных; проектирование интерфейса с пользователем. Структуры диалога; поддержка пользователя; многооконные интерфейсы. Объектное и событийное программирование, сравнительный анализ; Сравнение C++ с языками Java и C#.

#### *Требования к результатам освоения дисциплины*

Студент в соответствии с задачами профессиональной деятельности и целями основной образовательной программы после изучения дисциплины должен обладать следующими общекультурными, общепрофессиональными и профессиональными компетенциями:

способностью готовить конспекты и проводить занятия по обучению работников применению программно-методических комплексов, используемых на предприятии (ПК-4)

способностью участвовать в настройке и наладке программно-аппаратных комплексов (ОПК-4)

способностью осваивать методики использования программных средств для решения практических задач (ОПК-2)

#### *В результате освоения дисциплины студенты должны:*

**Знать:** инструментальные средства языков программирования и систем программирования;

**Уметь:** строить рациональные алгоритмы и на их основе создавать программные продукты на языках программирования C++ и C# в средах программирования Borland C++, Visual C++, Visual C#, как для решения сравнительно простых задач, так и для задач с функциями, структурами данных, указателями, строками и классами;

**Владеть:** технологией структурного и объектно-ориентированного программирования, а также средствами отладки.