

Факультет компьютерных технологий и прикладной математики

Направление и код подготовки/специальности (профиль): 02.03.03 Математическое обеспечение и администрирование информационных систем (Технологии разработки программных систем) / ОФО

Наименование и код дисциплины: Б1.О.13«Основы программирования»	
Количество академических часов (аудиторные/внеаудиторные): 118/91,5	Количество зачетных единиц: 6
Предварительные требования для изучения дисциплины: нет	Уровень подготовки: бакалавриат
Язык обучения: русский	Вид занятий по дисциплине: лекции – 50 ак.час., лабораторные занятия – 68 ак.час., самостоятельная работа – 56 ак.час
Курс/семестр: 1/осенний	Вид аттестации: осенний семестр – зачет, экзамен
Образовательные технологии: коммуникативного обучения, разноуровневого (дифференцированного) обучения, модульного обучения, информационно-коммуникационные технологии, использования компьютерных программ, Интернет-технологии, проектная технология, игровая технология, развития критического мышления.	
Краткая аннотация к содержанию дисциплины: формирование у студентов базовых знаний в области современного программирования, включая основные понятия, принципы разработки алгоритмов, их реализацию на языке программирования, а также методы тестирования и отладки программного кода.	
Темы лекционных и семинарских занятий: 1. Языки программирования. Основы алгоритмизации 2. Лексические основы языка 3. Скалярные типы. Выражения. Операции 4. Ввод/вывод 5. Условный оператор 6. Операторы цикла 7. Статические массивы 8. Указатели 9. Динамические массивы 10. Функции, прототипы функций 11. Структуры 12. Алгоритмы сортировки	
Полученные компетенции: Знать: • Основные синтаксические конструкции C++ (лексика, типы данных, операции, управляющие структуры). • Принципы работы с памятью (указатели, ссылки, статические и динамические массивы). • Алгоритмы обработки данных (сортировка, поиск) и их реализацию на C++. • Различные подходы к реализации алгоритмов (итеративные, рекурсивные методы). Уметь: • Анализировать условия задачи и выбирать подходящие языковые конструкции. • Применять условные операторы и циклы для управления потоком выполнения. • Работать с массивами, указателями и структурами данных. • Сравнить альтернативные варианты решения (например, алгоритмы сортировки) и обосновывать выбор. Владеть:	

- Навыками написания, отладки и тестирования программ на C++.
- Методами проектирования модульных программ (функции, структуры).
- Техниками оптимизации кода (эффективность по памяти и времени выполнения).