

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сыров Игорь Анатольевич
Должность: Директор
Дата подписания: 22.08.2023 10:52:37
Уникальный программный ключ:
b683afe664d7e9f64175886cf9626a196149ad36

СТЕРЛИТАМАКСКИЙ ФИЛИАЛ
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Факультет Математики и информационных технологий
Кафедра Математического моделирования

Аннотация рабочей программы дисциплины (модуля)

дисциплина **Математическое программирование**

Блок Б1, вариативная часть, Б1.В.ДВ.01.02

цикл дисциплины и его часть (базовая, вариативная, дисциплина по выбору)

Направление

10.03.01

Информационная безопасность

код

наименование направления

Программа

Безопасность компьютерных систем (по отрасли или в сфере профессиональной деятельности)

Форма обучения

Очная

Для поступивших на обучение в
2020 г.

1. Перечень планируемых результатов обучения по дисциплине (модулю)

1.1. Перечень планируемых результатов освоения образовательной программы

Выпускник, освоивший программу высшего образования, в рамках изучаемой дисциплины, должен обладать компетенциями, соответствующими видам профессиональной деятельности, на которые ориентирована программа:

| |
|--|
| Способностью применять соответствующий математический аппарат для решения профессиональных задач (ОПК-2) |
| Способностью проводить эксперименты по заданной методике, обработку, оценку погрешности и достоверности их результатов (ПК-11) |
| Способен использовать математические методы обработки, анализа и синтеза результатов профессиональных исследований (ПСК1-2) |

1.2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

| Формируемая компетенция (с указанием кода) | Этапы формирования компетенции | Планируемые результаты обучения по дисциплине (модулю) |
|--|---|---|
| Способностью проводить эксперименты по заданной методике, обработку, оценку погрешности и достоверности их результатов (ПК-11) | 1 этап: Знания | Обучающийся должен знать: методики обработки и оценки погрешностей полученных результатов в ходе экспериментальных исследований. |
| | 2 этап: Умения | Обучающийся должен уметь: делать выводы на основании полученных экспериментальных исследований, проводить дальнейшие пути корректировки. |
| | 3 этап: Владения (навыки / опыт деятельности) | Обучающийся должен владеть: приемами и методами решения конкретных задач из различных областей технологии, приемами обработки и оценки результатов экспериментальных исследований. |
| Способен использовать математические методы обработки, анализа и синтеза результатов профессиональных исследований (ПСК1-2) | 1 этап: Знания | Обучающийся должен знать: принципы использования программных пакетов для осуществления научных исследований; основные методы применения информационных технологий при проведении научных исследований; возможности различного вида математического аппарата как инструмента познания мира; технологию построения математических моделей систем различной природы; основные численные методы математических задач. |

| | | |
|--|---|--|
| | 2 этап: Умения | Обучающийся должен уметь: осуществлять математическую и информационную постановку задач; самостоятельно разрабатывать и реализовывать алгоритмы для решения научно-исследовательских задач; использовать основные законы естественнонаучных дисциплин, применять методы математического анализа и моделирования, теоретического и экспериментального исследования для анализа сложных систем и прогнозирования их поведения; - решать прикладные математические задачи с использованием современных инструментальных средств; моделировать процессы и анализировать модели с использованием информационных технологий; применять математические методы (в том числе численные) при решении профессиональных задач. |
| | 3 этап: Владения (навыки / опыт деятельности) | Обучающийся должен владеть: навыками выбора подходящих методов решения прикладных задач; методами математического анализа, корректной компьютерной обработки и последующего анализа результатов математического моделирования; методами решения задач, теоретического и экспериментального исследования. |
| Способностью применять соответствующий математический аппарат для решения профессиональных задач (ОПК-2) | 1 этап: Знания | Обучающийся должен знать: основные алгоритмические конструкции и основные операторы языков программирования C# и C++; основные структуры данных, способы их создания и обработки; основы структурного, модульного и визуального программирования; основные этапы разработки алгоритмов и программ; современные языки программирования и пакеты программ в области программирования; современные способы и средства работы с информацией для изучения основных конструкций современных языков программирования; общие принципы и навыки практического применения объектно-ориентированного программирования. |
| | 2 этап: Умения | Обучающийся должен уметь: выбирать |

| | | |
|--|---|--|
| | | структуры данных, необходимые для решения поставленной задачи; составлять алгоритмы обработки данных; разрабатывать программы на языках программирования С# и С++, проводить их отладку и тестирование; ориентироваться в постановках задач, при решении поставленных задач обоснованно строить алгоритмы, реализовывать их на языках программирования С# и С++. |
| | 3 этап: Владения (навыки / опыт деятельности) | Обучающийся должен владеть: методами использования в профессиональной деятельности языков программирования С# и С++; современными методами практического программирования конкретных задач в определенной языковой среде; умениями и навыками использования библиотек объектов (классов) для решения практических задач; современными методами и приемами эффективной обработки информации средствами языков программирования. |

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина реализуется в рамках вариативной части.

Для освоения дисциплины необходимы компетенции, сформированные в рамках изучения следующих дисциплин: «Информационные технологии», «Технологии и методы программирования», «Математическая логика и теория алгоритмов», «Моделирование бизнес-процессов».

Дисциплина «Математическое программирование» занимает важное место среди изучаемых дисциплин. В процессе работы студенты должны на основе изученных тем и рассмотренных примеров приобрести практические навыки и умения в конструировании программ, предназначенных для решения различных прикладных задач, в том числе связанных с информационными процессами и системами.

Дисциплина изучается на 3 курсе в 5 семестре

3. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 2 зач. ед., 72 акад. ч.

| Объем дисциплины | Всего часов |
|------------------|-------------|
|------------------|-------------|

| | |
|--|-----------------------------|
| | Очная форма обучения |
| Общая трудоемкость дисциплины | 72 |
| Учебных часов на контактную работу с преподавателем: | |
| лекций | 12 |
| практических (семинарских) | 20 |
| другие формы контактной работы (ФКР) | 0,2 |
| Учебных часов на контроль (включая часы подготовки): | |
| зачет | |
| Учебных часов на самостоятельную работу обучающихся (СР) | 39,8 |

| | |
|-----------------------|-----------------|
| Формы контроля | Семестры |
| зачет | 5 |

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)

| № п/п | Наименование раздела / темы дисциплины | Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость (в часах) | | | |
|----------|---|---|-----------|----------|-------------|
| | | Контактная работа с преподавателем | | | СР |
| | | Лек | Пр/Сем | Лаб | |
| 1 | Основные приёмы программирования на языке C++. Структуры данных. | 7 | 12 | 0 | 24 |
| 1.1 | Введение в язык C++. | 1 | 2 | 0 | 4 |
| 1.2 | Управляющие структуры | 2 | 2 | 0 | 4 |
| 1.3 | Массивы и указатели. | 1 | 2 | 0 | 4 |
| 1.4 | Строки в C++. | 1 | 2 | 0 | 4 |
| 1.5 | Функции как средство структуризации программы. | 1 | 2 | 0 | 4 |
| 1.6 | Динамические структуры. | 1 | 2 | 0 | 4 |
| 2 | Объектно-ориентированное программирование на C++. | 5 | 8 | 0 | 15,8 |
| 2.1 | Классы в C++. | 1 | 2 | 0 | 4 |
| 2.2 | Разработка структуры классов. | 1 | 1 | 0 | 4 |
| 2.3 | Дружественные функции и классы. | 1 | 1 | 0 | 4 |
| 2.4 | Иерархия классов. | 1 | 1 | 0 | 2 |
| 2.5 | Усложнение структуры классов. | 1 | 3 | 0 | 1,8 |
| | Итого | 12 | 20 | 0 | 39,8 |

4.2. Содержание дисциплины, структурированное по разделам (темам)

Курс лекционных занятий

| № | Наименование раздела / темы дисциплины | Содержание |
|----------|---|------------|
| 1 | Основные приёмы программирования на языке C++. Структуры данных. | |

| | | |
|----------|--|--|
| 1.1 | Введение в язык C++. | Алфавит, синтаксис, семантика языка. Структура программы. Простейшие операторы ввода-вывода. Порядковые типы. Целые, логические, символьные типы, их физическое представление. Другие типы. Вещественные типы, их физическое представление. Описание констант и переменных стандартных типов. Приведение типов. Выражения. |
| 1.2 | Управляющие структуры | Полная и неполная форма ветвления. Множественное ветвление. Оператор безусловного перехода goto. Циклы с предусловием и с постусловием. Оператор цикла с параметром. Операторы break и continue. Организация цикла с помощью рекурсивных процедур. |
| 1.3 | Массивы и указатели. | Адреса и указатели. Объявление указателей. Операции над указателями. Регулярные типы (одномерные и многомерные массивы), комбинированные типы их физическое представление. |
| 1.4 | Строки в C++. | Физическое представление строковых величин, операции со строками. Тип struct в языке программирования C++. |
| 1.5 | Функции как средство структуризации программы. | Описание функции и ее вызов. Локальные переменные. Способы передачи параметров функций (по значению и по адресу). Фактические и формальные параметры. Ссылки и ссылочные параметры. Функции с переменным количеством параметров. Прототипы функций. Перегрузка функций. Шаблоны функций. |
| 1.6 | Динамические структуры. | Конструирование объектов: динамические массивы, стеки, списки, очереди. |
| 2 | Объектно-ориентированное программирование на C++. | |
| 2.1 | Классы в C++. | Основные понятия. Классы. Конструкторы и деструкторы. Область действия и доступ к членам класса. |
| 2.2 | Разработка структуры классов. | Статические поля. Конструктор копирования. Указатель this. Передача значения по ссылке. |
| 2.3 | Дружественные функции и классы. | Организация взаимодействия нескольких объектов разных классов посредством дружественных функций. Взаимодействие классов на примере описания структуры «Клиент – Банк». |
| 2.4 | Иерархия классов. | Инкапсуляция. Вложенные классы в C++. Наследование и полиморфизм. Множественное наследование. Виртуальные методы. |
| 2.5 | Усложнение структуры классов. | Абстрактные классы. Виртуальные базовые классы. Шаблоны классов. Нетипизированные аргументы. Частичная специализация шаблона. |

Курс практических/семинарских занятий

| № | Наименование раздела / темы дисциплины | Содержание |
|----------|---|--|
| 1 | Основные приёмы программирования на языке C++. Структуры данных. | |
| 1.1 | Введение в язык C++. | Структура программы. Описание переменных. Инструкции ввода-вывода. Использование ветвления при решении задач, содержащих разнообразные условия. Программирование составных условий. Инструкция switch. |
| 1.2 | Управляющие структуры | Изучение циклов с параметром, с предусловием и с после условием. Применения каждого вида циклов в зависимости |

| | | |
|----------|--|--|
| | | от требований конкретной задачи. Вложенные циклы. Инструкции break и continue. |
| 1.3 | Массивы и указатели. | Указатели. Операции над указателями. Одномерные и многомерные массивы. Использование указателей при работе с массивами. |
| 1.4 | Строки в C++. | Необходимость введения составного типа данных. Составление сложных структур данных на основе типа struct. Примеры использования типа struct. |
| 1.5 | Функции как средство структуризации программы. | Использование функций при написании программ. Передача массивов в функции. Создание меню программы на основе функций. Эффективное использование функций. Параметры по умолчанию и перегрузка функций. Прототипы. |
| 1.6 | Динамические структуры. | Особенности работы с символами и строками в C++. Классовый тип string. Организация ввода-вывода между программой и объектом string. |
| 2 | Объектно-ориентированное программирование на C++. | |
| 2.1 | Классы в C++. | Классы для описания простых математических и геометрических объектов. Инкапсуляция. Примеры разработки классов для различных областей использования. |
| 2.2 | Разработка структуры классов. | Организация взаимодействия нескольких объектов разных классов посредством дружественных функций. Взаимодействие классов на отдельных примерах. |
| 2.3 | Дружественные функции и классы. | Механизм наследования. Множественное наследование. Конструкторы и деструкторы базовых и производных классов. Раннее и позднее связывание, виртуальные функции. |
| 2.4 | Иерархия классов. | Назначение и использование абстрактных классов. Абстрактные классы как основа более сложных структурных объектов. Невозможность создания экземпляров объектов абстрактных классов. Чисто виртуальные функции. |
| 2.5 | Усложнение структуры классов. | Механизм перегрузки операций. Перегрузка унарных и бинарных операций. Перегрузка операции вызова функции и операции индексирования. Перегрузка операций «поместить в поток» и «взять из потока». |