

СТЕРЛИТАМАКСКИЙ ФИЛИАЛ
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет
Кафедра

Математики и информационных технологий
Прикладной информатики и программирования

Аннотация рабочей программы дисциплины (модуля)

дисциплина

Б1.О.21 Технологии и методы программирования

обязательная часть

Направление

10.03.01

Информационная безопасность

код

наименование направления

Программа

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

Форма обучения

Очно-заочная

Для поступивших на обучение в
2021 г.

Стерлитамак 2022

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций

Формируемая компетенция (с указанием кода)	Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине (модулю)
ОПК-7. Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности;	ОПК-7.1. Знает языки программирования и системы разработки программных средств для решения профессиональных задач.	Обучающийся должен: знать современные компьютерные технологии и программные средства, используемые при разработке ПО; знать основные принципы создания программ на изучаемом языке программирования; синтаксис и семантику изучаемого языка программирования;
	ОПК-7.2. Способен выбирать необходимые языки программирования и системы разработки программных средств для решения профессиональных задач.	Обучающийся должен: применять на практике полученные знания и использовать доступные программные средства для создания программ; транслировать текстовую задачу в программный код; искать альтернативные пути решения, если для решения поставленной задачи не хватает инструментов языка;
	ОПК-7.3. Обладает навыками программирования и разработки систем программных средств для решения профессиональных задач.	Обучающийся должен: владеть навыками основ работы в известных программных оболочках для решения прикладных задач путем написания программ; владеть основными понятиями и принципами, использующимися в программировании; владеть навыками написания программ;

2. Цели и место дисциплины (модуля) в структуре образовательной программы

Цели изучения дисциплины:

Дисциплина реализуется в рамках базовой части. Для освоения дисциплины необходимы знания, полученные в рамках школьного курса информатики, а также курса «Алгоритмы и языки программирования» .

Дисциплина «Технологии и методы программирования» 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 для ЭВМ, предназначенных для решения различных прикладных задач, в том числе связанных с информационными процессами и системами.

Дисциплина изучается на 2 курсе в 4 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 4 зач. ед., 144 акад. ч.

Объем дисциплины	Всего часов
	Очно-заочная обучения
Общая трудоемкость дисциплины	144
Учебных часов на контактную работу с преподавателем:	
лекций	16
практических (семинарских)	32
другие формы контактной работы (ФКР)	3,2
Учебных часов на контроль (включая часы подготовки):	34,8
курсовая работа	
экзамен	
Учебных часов на самостоятельную работу обучающихся (СР):	58
курсовая работа	

Формы контроля	Семестры
курсовая работа	4
экзамен	4

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	
1.3	Дружественные функции и классы.	2	4	0	10
1.2	Разработка структуры классов.	2	6	0	10
1.1	Классы в С++.	4	8	0	10
1	Объектно-ориентированное программирование на С++.	16	32	0	58
1.5	Усложнение структуры классов.	2	2	0	10
1.4	Иерархия классов.	2	8	0	10
1.6	Перегрузка операций.	4	4	0	8
	Итого	16	32	0	58

4.2. Содержание дисциплины, структурированное по разделам (темам)

Курс лекционных занятий

№	Наименование раздела / темы дисциплины	Содержание
1.3	Дружественные функции и классы.	Организация взаимодействия нескольких объектов разных классов посредством дружественных функций. Взаимодействие классов на примере описания структуры «Клиент – Банк».
1.2	Разработка структуры классов.	Статические поля. Конструктор копирования. Указатель this. Передача значения по ссылке.
1.1	Классы в C++.	Основные понятия. Классы. Конструкторы и деструкторы. Область действия и доступ к членам класса.
1	Объектно-ориентированное программирование на C++.	
1.5	Усложнение структуры классов.	Абстрактные классы. Виртуальные базовые классы. Шаблоны классов. Нетипизированные аргументы. Частичная специализация шаблона
1.4	Иерархия классов.	Инкапсуляция. Вложенные классы в C++. Наследование и полиморфизм. Множественное наследование. Виртуальные методы.
1.6	Перегрузка операций.	Механизм перегрузки операций. Перегрузка унарных и бинарных операций. Перегрузка операции вызова функции и операции индексирования. Перегрузка операций «поместить в поток» и «взять из потока».

Курс практических/семинарских занятий

№	Наименование раздела / темы дисциплины	Содержание
1.3	Дружественные функции и классы.	Механизм наследования. Множественное наследование. Конструкторы и деструкторы базовых и производных классов. Раннее и позднее связывание, виртуальные функции
1.2	Разработка структуры классов.	Организация взаимодействия нескольких объектов разных классов посредством дружественных функций. Взаимодействие классов на отдельных примерах.
1.1	Классы в C++.	Классы для описания простых математических и геометрических объектов. Инкапсуляция. Примеры разработки классов для различных областей использования.
1	Объектно-ориентированное программирование на C++.	
1.5	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».
1.4	Иерархия классов.	Назначение и использование абстрактных классов. Абстрактные классы как основа более сложных структурных объектов. Невозможность создания экземпляров объектов абстрактных классов. Чисто виртуальные функции.
1.6	Перегрузка операций.	Практическое применение и обобщение приемов и методов разработки классов для описания математических и других (физических, социальных, и т.д.) объектов.

