

Министерство науки и высшего образования РФ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.О.08.15 МОДУЛЬ ДИСЦИПЛИН ГИБРИДНОГО
ФОРМАТА

Программирование на языке C++

наименование дисциплины (модуля) в соответствии с учебным планом

Направление подготовки / специальность

09.03.02 Информационные системы и технологии

Направленность (профиль)

09.03.02.31 Разработка компьютерных игр и приложений

Форма обучения

очная

Год набора

2023

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Программу составили _____

ассистент, Халтурин Евгений Александрович

должность, инициалы, фамилия

1 Цели и задачи изучения дисциплины

1.1 Цель преподавания дисциплины

Рассмотрение возможностей языка программирования C++, его особенностей, сильных и слабых сторон. Формирование навыков разработки эффективных приложений на языке C++.

1.2 Задачи изучения дисциплины

- Научить основам языка программирования C++. Показать продвинутое возможности языка программирования, его нововведения, добавленные стандартом 14, 17 и 20 года.

- Научить выстраивать правильные причинно-следственные связи.

- Сформировать умение использования отладчика для исправления ошибок при проектировании программ.

- Сформировать умение оценивать время выполнения программ, их затраты по памяти.

- Получение студентами навыков использования структуры данных для решения прикладных задач.

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

Код и наименование индикатора достижения компетенции	Запланированные результаты обучения по дисциплине
ОПК-6: Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий;	
ОПК-6.1: Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ	Знать основные типы переменных языка C++. Знать структуры данных, реализованные в STL. Знать структуры данных, которые отсутствуют в STL (Двоичное дерево, Декартовое дерево). Уметь использовать условия и циклы. Уметь использовать рекурсивные подходы. Уметь использовать современные подходы проектирования и алгоритмизации. Владение базовыми структурами данных STL. Владение продвинутыми структурами данных STL. Владение продвинутыми структурами данных, в том числе и из сторонних библиотек.

ОПК-6.2: Программирует, осуществляет отладку и тестирование прототипов программно-технических комплексов	Знание основных способов отладки программного кода. Знания модульного тестирования. Знание сторонних программных утилит для тестирования кода. Уметь определять асимптотику времени и памяти.
	Уметь определять время выполнения программы. Уметь предотвращать гонку за ресурсы в многопоточных приложениях. Владение условными точками останова. Владение условной компиляцией. Владение метапрограммированием с использованием шаблонных типов.

1.4 Особенности реализации дисциплины

Язык реализации дисциплины: Русский.

Дисциплина (модуль) реализуется с применением ЭО и ДОТ

URL-адрес и название электронного обучающего курса: Дисциплина реализуется исключительно в ЭО и ДОТ. Ссылка на электронный курс: <https://e.sfu-kras.ru/course/view.php?id=36669>.

2. Объем дисциплины (модуля)

Вид учебной работы	Всего, зачетных единиц (акад. час)	е
		1
Контактная работа с преподавателем:	2 (72)	
занятия лекционного типа	1 (36)	
практические занятия	1 (36)	
Самостоятельная работа обучающихся:	2 (72)	
курсовое проектирование (КП)	Нет	
курсовая работа (КР)	Нет	

3 Содержание дисциплины (модуля)

3.1 Разделы дисциплины и виды занятий (тематический план занятий)

		Контактная работа, ак. час.							
№ п/п	Модули, темы (разделы) дисциплины	Занятия лекционного типа		Занятия семинарского типа				Самостоятельная работа, ак. час.	
				Семинары и/или Практические занятия		Лабораторные работы и/или Практикумы			
		Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС
1. Введение, типы данных, операции над ними									
	1. Начало написания кода на языке C++. Переменные. Библиотеки.	2	2						
	2. Операции языка C++.	2	2						
	3. Простейшие программы. Знакомство с выбранной средой разработки.			2	2				
	4. Работа с консолью и файлами. Использование различных способов ввода\вывода. Оценка времени вывода.			2	2				
2. Условия, циклы, работа с массивами									
	1. Разветвление выполнения программы. Условия. Множественные условия. Циклы, их вариации	2	2						
	2. Тернарный оператор условия. Возвращаемое значение, приведение типов.	2	2						

3. Оператор goto, метки. Статические и динамические массивы. STD vector и string, способ использования.	2	2						
4. Написание текстовой игры при помощи проверки условий.			2	2				
5. Выполнение заданий на изучение арифметических, логических и битовых операций.			2	2				
6. Указатели, ссылки. Хранение переменных в памяти. Область видимости переменных.			2	2				
7. Массивы. Статический и динамический вариант. Время жизни объекта.			2	2				
8. Работа с массивами, анализ данных.			2	2				
3. Функции, рекурсия								
1. Синтаксис функций.	2	2						
2. Способы передачи параметров.	2	2						
3. Рекурсивный вызов функций. Выход из рекурсии. Оптимизация. pragma. Макросы.			2	2				
4. Написание простой консольной (файловой) игры, например, морской бой, крестики-нолики и другие.			2	2				
4. Классы, шаблонный параметр								
1. Структура и классы, их экземпляры.	2	2						
2. Шаблонные функции и структуры. Тип auto.	2	2						
3. Рефакторинг кода. Сокращение кода, объединение логических элементов программы в функции.			2	2				
5. Алгоритмы и структуры данных языка C++.								
1. Библиотека STD.	2	2						
2. Реализованные структуры данных, их сравнение.	2	2						

3. Библиотека <algorithm>. Основные функции. Стандартный интерфейс структур данных.			2	2				
4. Использование STD. Решение стандартных задач с использованием STD.			2	2				
5. Использование собственных структур данных. Асимптотическая оценка скорости выполнения и затрат памяти.			2	2				
6. ООП в C++								
1. ООП. Принципы ООП.	2	2						
2. Модификаторы доступа, наследование, инкапсуляция данных. Пространство имён.	2	2						
3. Проектирование в Unreal Engine 5.	2	2						
4. Написание генератора карт.			2	2				
5. Написание искусственного интеллекта (бота) для своего приложения.			2	2				
7. Оптимизации и интерфейсы языка C++								
1. Итераторы. Компараторы. Составные типы данных. Кортежи, Композиция и декомпозиция данных. void*.	2	2						
2. Умные указатели. Семантика перемещения. Подстановка функций.	2	2						
3. Решение задач с использованием итераторов. Задачи с применением собственного компаратора.			2	2				
8. Работа с графикой								
1. Графические библиотеки. SFML.	2	2						
2. Unreal Engine 5.	2	2						
3. Разработка расширяемого архитектурного решения программы.			2	2				

4. Добавление графики в приложении.			2	2				
5. Самостоятельная работа. Написание консольной игры							72	72
Всего	36	36	36	36			72	72

4 Учебно-методическое обеспечение дисциплины

4.1 Печатные и электронные издания:

1. Павловская Т. А. С/С++. Программирование на языке высокого уровня: учебник для студентов вузов по направлению "Информатика и вычислительная техника": допущено Министерством образования РФ (Санкт-Петербург: Питер).
2. Павловская Т. А. С/С++. Программирование на языке высокого уровня: учебник для вузов (магистров и бакалавров)(Москва: Питер).

4.2 Лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства (программное обеспечение, на которое университет имеет лицензию, а также свободно распространяемое программное обеспечение):

1. Microsoft Visual Studio

4.3 Интернет-ресурсы, включая профессиональные базы данных и информационные справочные системы:

1. Не требуется.

5 Фонд оценочных средств

Оценочные средства находятся в приложении к рабочим программам дисциплин.

6 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине (модулю)

Дисциплина реализуется в полностью с применением ЭО и ДОТ. Соответственно, для обучения студенту требуется стационарный компьютер или ноутбук с устойчивым подключением к информационно-телекоммуникационной сети «Интернет», доступом к сервисам ЭИОС СФУ. Взаимодействие студента с руководителем осуществляется через сервисы видеоконференций (синхронное) и сервисы ЭИОС СФУ (асинхронное). Однако при необходимости студенты могут воспользоваться материально-технической базой университета, соответствующей действующим санитарным и противопожарным правилам и нормам.