



SCHOOL OF BUSINESS
AND MANAGEMENT OF
TECHNOLOGY OF BSU



Innovative ICT Education for Social-Economic Development (IESED)
574283-EPP-1-2016-1-LT-EPPKA2-CBHE-JP

Государственное учреждение образования
«Институт бизнеса БГУ»

УТВЕРЖДАЮ

Директор

_____ П.И. Бригадин

" ____ " _____

Экспериментальная учебная программа (по проекту Erasmus+)

по дисциплине

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ
для специальности

1-58 01 01 Управление информационными ресурсами

СОСТАВИТЕЛИ:

Силкович Ю.Н., кандидат технических наук, доцент

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой цифровых систем и технологий Государственного учреждения образования
«Институт бизнеса БГУ»

(протокол № __ от _____);

Заведующий кафедрой

Силкович Ю.Н.

1. ПЛАН КУРСА

Год обучения	Семестр	Академические часы					Курсовая работа	Зачётные единицы	Форма обучения
		Всего	Лекции	Лабораторные	Практика / семинары	Независимая работа			
1	1,2	190	52	68	-	70	-	7	очная
1	1,2	190	16	20	-	154	-	7	заочная

2. КОМПЕТЕНЦИИ

1. Разрабатывать структуры данных для использования в информационных системах, системах операционного анализа и интеллектуальных системах.
2. Моделировать, разрабатывать программное обеспечение, и техническую документацию для поддержки деятельности в различных предметных областях.
3. Проводить всестороннее тестирование разработанных программных продуктов и прикладного программного обеспечения.

3. ЦЕЛИ КУРСА

Подготовка специалиста с фундаментальными знаниями и практическими навыками в области алгоритмов и программирования.

4. РЕЗУЛЬТАТЫ

После завершения этого курса студент сможет:

- разрабатывать структуры данных для решения прикладных задач.
- разрабатывать алгоритмы для решения реальных проблем.
- использовать разработки приложений программного обеспечения для проектирования систем.
- применять соответствующие методы тестирования и отладки.

5. УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА (Для очного обучения)

Номер раздела, темы, занятия	Название раздела, темы, занятия; перечень изучаемых вопросов	Количество академических часов				Форма контроля знаний
		лекция	практические (семинарские) занятия	лабораторные занятия	управляемая самостоятельная работа студента	
1.	Основы алгоритмизации	3	3	0	0	Итоговый экзамен
1.1.	Понятие и свойства алгоритма.	1	1			
1.2.	Методы описания алгоритмов	1	1			
1.3.	Этапы разработки программы	1	1			
2.	C ++ Основы языка	18,5	6,5	12	0	Итоговый экзамен
2.1.	C ++ базовый синтаксис	0,5	0,5			
2.2.	Типы данных в C ++	1	1			
2.3.	Переменные, константы в C ++	0,5	0,5			
2.4.	Структура исходного кода в C ++.	0,5	0,5			
2.5.	Операции в C ++	1	1			
2.6.	Функции ввода/вывода в C ++.	3	1	2		
2.7.	Операторы в C ++	12	2	10		Лабораторные работы
3.	Основы # языка C	18,5	6,5	12	0	Итоговый экзамен
3.1.	C #. Основной синтаксис	0,5	0,5			
3.2.	Типы данных в C #	1	1			
3.3.	Структура исходного кода в C #.	0,5	0,5			
3.4.	Переменные, константы в C #	0,5	0,5			
3.5.	Операции	1	1			
3.6.	Использование классов ввода / вывода FCL.	3	1	2		
3.7.	Операторы в C #	12	2	10		Лабораторные работы
4.	Структурированные типы данных	28	10	18	0	Итоговый экзамен
4.1.	Массивы в C ++	3	1	2		
4.2.	Структуры в C ++	3	1	2		
4.3.	Класс массива C #	3	1	2		
4.4.	Структуры в C #	3	1	2		
4.5.	Алгоритмы обработки массивов.	16	6	10		Лабораторные работы

5.	Указатели и ссылки в C ++	7	3	4	0	Итоговый экзамен
5.1.	Указатели	1	1			
5.2.	Распределение динамической памяти.	6	2	4		
6.	функции	14	6	8	0	Итоговый экзамен
6.1.	Функции в C ++.	1	1			
6.2.	Передача параметров в функции в C ++.	3	1	2		
6.3.	Функции и процедуры в C #	1	1			
6.4.	Передача параметров в функции в C #.	3	1	2		
6.5.	Рекурсивные вызовы и их приложения.	6	2	4		Лабораторные работы
7.	Файлы и потоки	18	4	14	0	Итоговый экзамен
7.1.	Понятие файла и потока в C ++.	1	1			
7.2.	Стандартные функции для работы с файлами в C ++.	3	1	2		
7.3.	Понятие потока в C #.	3	1	2		
7.4.	Использование классов потоков.	3	1	4		Лабораторные работы
7.5.	Алгоритмы обработки файлов.	6		6		Лабораторные работы
8.	Строки	15	5	10	0	Итоговый экзамен
8.1.	Массивы символов в C ++	1	1			
8.2.	C ++ функции обработки символьных массивов	3	1	2		
8.3.	Использование класса String, в C #.	3	1	2		
8.4.	Алгоритмы обработки строк	8	2	6		Лабораторные работы
9.	Классы и объекты в C ++	23	9	14	0	Итоговый экзамен
9.1.	Объектная парадигма	1	1			
9.2.	Классы	6	2	4		Лабораторные работы
9.3.	наследование	6	2	4		Лабораторные работы
9.4.	Динамические объекты.	4	2	2		Лабораторные работы
9.5.	Виртуальные методы.	6	2	4		Лабораторные работы
10.	Динамические структуры данных	37	11	26	0	
10.1	Общие принципы организации динамических структур в C ++.	1	1			
10.2	Алгоритмы для	10	2	8		Лабораторные

	двунаправленного обработки списков.					работы
10.3	Алгоритмы Binary обработки деревьев.	14	4	10		Лабораторные работы
10.4	Алгоритмы обработки графов.	12	4	8		Лабораторные работы
11.	Шаблоны классов и функций в C ++	8	2	6	0	Решающий Итоговый экзамен
11.1	Шаблоны функций.	1	1			
11.2	Шаблоны классов.	7	1	6		Лабораторные работы
	ВСЕГО	190	66	124	0	

6. УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА (Для заочного обучения)

Номер раздела, темы, занятия	Название раздела, темы, занятия; перечень изучаемых вопросов	Количество академических часов				Форма контроля знаний
		лекция	практические (семинарские) занятия	лабораторные занятия	управляемая самостоятельная работа студента	
1.	Основы алгоритмизации	3	0,3	0	2,7	Итоговый экзамен
1.1.	Понятие и свойства алгоритма.	1	0,1		0,9	
1.2.	Методы описания алгоритмов	1	0,1		0,9	
1.3.	Этапы разработки программы	1	0,1		0,9	
2.	C ++ Основы языка	18,5	1,4	2	15,1	Итоговый экзамен
2.1.	C ++ базовый синтаксис	0,5	0,1		0,4	
2.2.	Типы данных в C ++	1	0,2		0,8	
2.3.	Переменные, константы в C ++	0,5	0,1		0,4	
2.4.	Структура исходного кода в C ++.	0,5	0,1		0,4	
2.5.	Операции в C ++	1	0,2		0,8	
2.6.	Функции ввода/вывода в C ++.	3	0,2		2,8	
2.7.	Операторы в C ++	12	0,5	2	9,5	Лабораторные работы
3.	Основы # языка C	18,5	1,4	2	17,1	Итоговый экзамен
3.1.	C #. Основной синтаксис	0,5	0,1		0,4	
3.2.	Типы данных в C #	1	0,2		0,8	
3.3.	Структура исходного кода в C #.	0,5	0,1		0,4	
3.4.	Переменные, константы в C #	0,5	0,1		0,4	
3.5.	Операции	1	0,2		0,8	
3.6.	Использование классов ввода /	3	0,2		2,8	

	вывода FCL.					
3.7.	Операторы в C #	12	0,5	2	11,5	Лабораторные работы
4.	Структурированные типы данных	28	3,2	4	20,8	Итоговый экзамен
4.1.	Массивы в C ++	3	0,5		2,5	
4.2.	Структуры в C ++	3	0,1		2,9	
4.3.	Класс массива C #	3	0,5		2,5	
4.4.	Структуры в C #	3	0,1		2,9	
4.5.	Алгоритмы обработки массивов.	16	2	2	10	Лабораторные работы
5.	Указатели и ссылки в C ++	7	0,3	0	6,7	Итоговый экзамен
5.1.	Указатели	1	0,1		0,9	
5.2.	Распределение динамической памяти.	6	0,2		5,8	
6.	Функции	14	1,6	2	10,4	Итоговый экзамен
6.1.	Функции в C ++.	1	0,1		0,9	
6.2.	Передача параметров в функции в C ++.	3	0,2		2,8	
6.3.	Функции и процедуры в C #	1	0,1		0,9	
6.4.	Передача параметров в функции в C #.	3	0,2		2,8	
6.5.	Рекурсивные вызовы и их приложения.	6	1	2	3	Лабораторные работы
7.	Файлы и потоки	18	1,7	2	14,3	Итоговый экзамен
7.1.	Понятие файла и потока в C ++.	1	0,2		0,8	
7.2.	Стандартные функции для работы с файлами в C ++.	3	0,2		2,8	
7.3.	Понятие потока в C #.	3	0,1		2,9	
7.4.	Использование классов потоков.	3	0,2		4,8	Лабораторные работы
7.5.	Алгоритмы обработки файлов.	6	1	2	3	Лабораторные работы
8.	Строки	15	1,5	2	11,5	Итоговый экзамен
8.1.	Массивы символов в C ++	1	0,1		0,9	
8.2.	C ++ функции обработки символьных массивов	3	0,2		2,8	
8.3.	Использование класса String, в C #.	3	0,2		2,8	
8.4.	Алгоритмы обработки строк	8	1	2	5	Лабораторные работы
9.	Классы и объекты в C ++	23	1,2	2	19,8	Итоговый экзамен
9.1.	Объектная парадигма	1	0,2		0,8	
9.2.	Классы	6	0,1	1	4,9	Лабораторные работы
9.3.	наследование	6	0,2	1	4,8	Лабораторные работы
9.4.	Динамические объекты.	4	0,2		3,8	Лабораторные работы

9.5.	Виртуальные методы.	6	0,5		5,5	Лабораторные работы
10.	Динамические структуры данных	37	3,1	6	27,9	
10.1.	Общие принципы организации динамических структур в C ++.	1	0,1		0,9	
10.2.	Алгоритмы для двунаправленной обработки списков.	10	1	2	7	Лабораторные работы
10.3.	Алгоритмы Binary обработки деревьев.	14	1	2	11	Лабораторные работы
10.4.	Алгоритмы обработки графов.	12	1	2	9	Лабораторные работы
11.	Шаблоны классов и функций в C ++	8	0,3	0	7,7	Решающий Итоговый экзамен
11.1.	Шаблоны функций.	1	0,1		0,9	
11.2.	Шаблоны классов.	7	0,2		6,8	Лабораторные работы
	ВСЕГО	190	16	20	154	

7. Теоретическое содержание

№	Названия тем	Содержание
1.	Основы алгоритмизации	
1.1.	Понятие и свойства алгоритма.	Определение алгоритма. Понимание свойств алгоритма.
1.2.	Методы описания алгоритмов	Вербальные-математическое описание. Блок-схемы. Основные принципы составления схем.
1.3.	Этапы разработки программы	Компиляция. Исполняемый код. Ошибки классификации. Отладка. Программирование парадигм.
2.	C ++ Основы языка	
2.1.	C ++ базовый синтаксис	Алфавит, лексемы и разделители, именование.
2.2.	Типы данных в C ++	Понятие, классификация, описание, совместимость типов, преобразование.
2.3.	Переменные, константы в C ++	Переменные: описание, инициализация. Символьные константы. Область видимости и время жизни. Const ключевое слово.
2.4.	Структура исходного кода в C ++.	Источник структура кода. Препроцессор понимание. Директивы. Утверждение.
2.5.	Операции в C ++	Понятие, классификация, приоритет, стандартные операции. Ветвление операторы.
2.6.	Функции ввода/вывода в C ++.	Форматирование, функций ввода / вывода. Строка, входные символы / выходные функции.
2.7.	Операторы в C ++	Отрасль, езда на велосипеде операторы. Итерационные расчеты. Итерационные расчеты с заданной точностью.
3.	Основы # языка C	
3.1.	C #. Основной синтаксис	Алфавит, лексемы и разделители.
3.2.	Типы данных в C #	Понятие, классификация, описание, совместимость типов
3.3.	Структура исходного	Структура исходного текста. Структура

	кода в C #.	операторов Модули.
3.4.	Переменные, константы в C #	Переменные: описание, инициализация. Символьные константы. Время жизни, область видимости.
3.5.	Операции	Понятие, классификация, приоритет, стандартные операции. Ветвление операторы. Итерационные расчеты с заданной точностью
3.6.	Использование классов ввода / вывода FCL.	Классы консольного ввода / вывода. Методы.
3.7.	Операторы в C #	Операторы ветвления, циклы. Итерационные расчеты. Итерационные расчеты с заданной точностью.
4.	Структурированные типы данных	
4.1.	Массивы в C ++	Концепция, декларация, инициализация. Доступ к элементам массива. Перечисления.
4.2.	Структуры в C ++	Архитектура структурного типа и особенностей использования в программе.
4.3.	Класс массива C #	Концепция, декларация, инициализация. Доступ к элементам массива. Перечисления.
4.4.	Структуры в C #	Архитектура структурного типа и особенностей использования в программе.
4.5.	Алгоритмы обработки массивов.	Ввод / вывод данных. Поисквые алгоритмы. Алгоритмы сортировки массивов. Алгоритмы обработки 2D-массивов.
5.	Указатели и ссылки в C ++	
5.1.	Указатели	Концепция, декларация, инициализация. Операции с указателями.
5.2.	Распределение динамической памяти.	Функции создания, удаления, перераспределения. Ссылки на массивы. Динамическая обработка массивов.
6.	Функции	
6.1.	Функции в C ++.	Декларация и реализация. Возвращаемое значение из функции.
6.2.	Передача параметров в функции в C ++.	Декларация и передача функции параметров значений и параметров-ссылок
6.3.	Функции и процедуры в C #	Декларация и реализация. Возвращаемое значение из функции.
6.4.	Передача параметров в функции в C #.	Декларация и передача функции параметра-значения, и параметры-ссылки. Использование служебных свойств ref, out.
6.5.	Рекурсивные вызовы и их приложения.	Метод рекурсивного программирования. Прикладное использование рекурсивных алгоритмов.
7.	Файлы и потоки	
7.1.	Понятие файла и потока в C ++.	Типы файлов, особенности структуры.
7.2.	Стандартные функции для работы с файлами в C ++.	Организация ввода-вывода с помощью потоковых классов CIN и COUT.
7.3.	Понятие потока в C #.	Пространство имен System.IO.

7.4.	Использование классов потоков.	Ввод / вывод символьных данных. Ввод / вывод двоичных данных. Доступ к данным файла. MemoryStream класс.
8.	Алгоритмы обработки файлов.	
8.1.	Строки	Концепция, декларация, инициализация.
8.2.	Массивы символов в C ++	Стандартная библиотека функций для обработки строк.
8.3.	C ++ функции обработки символьных массивов	Работа со строками в .Net, основы регулярных выражений.
8.4.	Использование класса String, в C #.	Алгоритмы поиска в тексте. Разбиение текста, замена, вставка, слияние
8.5.	Алгоритмы обработки строк	Алгоритмы для чтения, записи и изменения файлов.
9.	Классы и объекты в C ++	
9.1.	Объектная парадигма	Инкапсуляция, наследование, полиморфизм.
9.2.	Классы	Определение классов. Поля, методы, модификаторы доступа.
9.3.	наследование	Наследование классов. Наследование модификаторы. Наследование, перекрытие методов и полей
9.4.	Динамические объекты.	Конструкторы, деструкторы. Указатель this.
9.5.	Виртуальные методы.	Абстрактные классы. Виртуальные методы. Наследование виртуальных методов. Использование абстрактных классов.
10.	Динамические структуры данных	
10.1.	Общие принципы организации динамических структур в C ++.	Узлы динамических структур. Типы динамических структур.
10.2.	Алгоритмы для двунаправленного обработки списков.	Алгоритмы формирования списка, вставки, удаления, перестановки и поиска элементов.
10.3.	Алгоритмы Binary обработки деревьев.	Алгоритмы формирования поискового бинарного дерева. Вставка, удаление и поиск. Алгоритмы обхода дерева. Решение прикладных задач с использованием бинарных деревьев.
10.4.	Алгоритмы обработки графов.	Алгоритмы формирования графов. Алгоритмы обхода графа, поиска пути. Решение прикладных задач с использованием графов.
11.	Шаблоны классов и функций в C ++	
11.1.	Шаблоны функций.	Понятие шаблона. Определение Шаблоны функций. Использование шаблонов функций.
11.2.	Шаблоны классов.	Определение классов шаблона. Создание экземпляров шаблонных классов.

8. ЛАБОРАТОРНАЯ ПРАКТИКА

№	Названия тем	Содержание
2.	С ++ Основы языка	
2.6.	Функции ввода/вывода в С ++.	Создание простой программы для линейного расчета с форматированным выводом.
2.7.	Операторы в С ++	Создание программ для реализации разветвленных и циклических алгоритмов. Итерационные расчеты с заданной точностью.
3.	Основы языка С#	
3.6.	Использование классов ввода / вывода FCL.	Создание простой программы для линейного расчета с форматированным выводом.
3.7.	Операторы в С#	Создание программ для реализации разветвленности и циклических алгоритмов. Итерационные расчеты с заданной точностью.
4.	Структурированные типы данных	
4.1.	Массивы в С ++	Реализация массива элементов I / O алгоритмы.
4.2.	Структуры в С ++	Структура Моделирования данных для хранения таблицы информации.
4.3.	Класс массива С #	Реализация массива элементов I / O алгоритмы.
4.4.	Структуры в С #	Структура Моделирования данных для хранения таблицы информации.
4.5.	Алгоритмы обработки массивов.	Реализация поиска, алгоритмы сортировки для решения прикладных задач. Алгоритмы. Использование 2D массивов для моделирования 2D-объектов.
5.	Указатели и ссылки в С ++	
5.2.	Распределение динамической памяти.	Работа с динамическими массивами.
6.	функции	
6.2.	Передача параметров в функции в С ++.	Создание и использование функции.
6.4.	Передача параметров в функции в С #.	Создание и использование функции.
6.5.	Рекурсивные вызовы и их приложения.	Используя рекурсивный алгоритм для двумерного лабиринта.
7.	Файлы и потоки	
7.2.	Стандартные функции для работы с файлами в С ++.	Разработка пользовательского интерфейса для работы с файлами.
7.3.	Понятие потока в С #.	Разработка пользовательского интерфейса для работы с файлами.
7.4.	Использование классов потоков.	Работа с классами потоков FCL.
7.5.	Алгоритмы обработки файлов.	Реализация различных структур хранения данных в файл.
8.	Строки	
8.2.	С ++ функции обработки символьных	Выполнение простых символов по-характер операции со строковыми данными.

	массивов	
8.3.	Использование класса String, в C #.	Выполнение простых символов по-характер операции со строковыми данными.
8.4.	Алгоритмы обработки строк	Преобразование файлов с исходным кодом программы в C ++
9.	Классы и объекты в C++	
9.2.	Классы	Разработка 2D-геометрических фигур классов.
9.3.	наследование	Разработка 3D классов геометрических фигур путем расширения классов 2D фигур.
9.4.	Динамические объекты.	Разработка динамических структур для хранения экземпляров 3D форм классов.
9.5.	Виртуальные методы.	Выполнение стандартных операций на различных 3D-объектов.
10.	Динамические структуры данных	
10.2.	Алгоритмы для двунаправленного обработки списков.	Создание двунаправленного класса списка. Решение прикладных задач с использованием списка двунаправленного класса.
10.3.	Алгоритмы Binary обработки деревьев.	Обработка двоичных деревьев с помощью рекурсивных алгоритмов
10.4.	Алгоритмы обработки графов.	Решение прикладных задач с использованием графиков.
11.	Шаблоны классов и функций в C ++	
11.2.	Шаблоны классов.	Создание и использование шаблонного класса

9. ЗАДАНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

№	Задачи
1.	Тестирование в среде VS
2.	Создание проекта с несколькими модулями
3.	Реализация итеративных вычислений с контролем входных значений и итерационного процесса.
4.	Алгоритмы быстрой сортировки одномерных массивов.
5.	Алгоритмы нестандартного извлечения элементов двумерных массивов.
6.	Организация хранения таблицы на основе структур.
7.	Рекурсивный обработка двумерного массива.
8.	Разбор строк.
9.	Сортировка файлов.
10.	Развитие класса «однонаправленный список».
11.	Разработка класса «стек»
12.	Разработка «очередь» шаблоны классов с использованием
13.	Формирование вывода таблицы с помощью класса cout.
14.	Чтение из файла, используя классы потоков.

10. Система оценки знаний и навыков (в соответствии с национальными требованиями)

Десятибалльная шкала в зависимости от величины балла и отметки включает следующие критерии:

10 (десять) баллов, зачтено:

систематизированные, глубокие и полные знания по всем разделам учебной программы учреждения высшего образования по учебной дисциплине, а также по основным вопросам, выходящим за ее пределы;

точное использование научной терминологии (в том числе на иностранном языке), грамотное, логически правильное изложение ответа на вопросы;

безупречное владение инструментарием учебной дисциплины, умение его эффективно использовать в постановке и решении научных и профессиональных задач;

выраженная способность самостоятельно и творчески решать сложные проблемы в нестандартной ситуации;

полное и глубокое усвоение основной, дополнительной литературы, по изучаемой учебной дисциплине;

умение свободно ориентироваться в теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им аналитическую оценку, использовать научные достижения других дисциплин;

творческая самостоятельная работа на практических, лабораторных занятиях, активное творческое участие в групповых обсуждениях высокий уровень культуры исполнения заданий.

9 (девять) баллов, зачтено:

систематизированные, глубокие и полные знания по всем разделам учебной программы учреждения высшего образования по учебной дисциплине;

точное использование научной терминологии (в том числе на иностранном языке), грамотное, логически правильное изложение ответа на вопросы;

владение инструментарием учебной дисциплины, умение его эффективно использовать в постановке и решении научных и профессиональных задач;

способность самостоятельно и творчески решать сложные проблемы в нестандартной ситуации в рамках учебной программы учреждения высшего образования по учебной дисциплине;

полное усвоение основной и дополнительной литературы, рекомендованной учебной программой учреждения высшего образования по учебной дисциплине;

умение ориентироваться в теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им аналитическую оценку;

систематическая, активная самостоятельная работа на практических, лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

8 (восемь) баллов, зачтено:

систематизированные, глубокие и полные знания по всем разделам учебной программы учреждения высшего образования по учебной дисциплине в объеме учебной программы учреждения высшего образования по учебной дисциплине;

использование научной терминологии (в том числе на иностранном языке), грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы и обобщения;

владение инструментарием учебной дисциплины (методами комплексного анализа, техникой информационных технологий), умение его использовать в постановке и решении научных и профессиональных задач;

способность самостоятельно решать сложные проблемы в рамках учебной программы учреждения высшего образования по учебной дисциплине;

усвоение основной и дополнительной литературы, рекомендованной учебной программой учреждения высшего образования по учебной дисциплине;

умение ориентироваться в теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им аналитическую оценку;

активная самостоятельная работа на практических, лабораторных занятиях, систематическое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

7 (семь) баллов, зачтено:

систематизированные, глубокие и полные знания по всем разделам учебной программы учреждения высшего образования по учебной дисциплине;

использование научной терминологии (в том числе на иностранном языке), грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы и обобщения;

владение инструментарием учебной дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;

свободное владение типовыми решениями в рамках учебной программы учреждения высшего образования по учебной дисциплине;

усвоение основной и дополнительной литературы, рекомендованной учебной программой учреждения высшего образования по учебной дисциплине;

умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им аналитическую оценку;

самостоятельная работа на практических, лабораторных занятиях, участие в групповых обсуждениях, высокий) уровень культуры исполнения заданий.

6 (шесть) баллов, зачтено:

достаточно полные и систематизированные знания в объеме учебной программы учреждения высшего образования по учебной дисциплине;

использование необходимой научной терминологии, грамотное, логически правильное изложение ответа на вопросы, умение делать обобщения и обоснованные выводы;

владение инструментарием учебной дисциплины, умение его использовать в решении учебных и профессиональных задач;

способность самостоятельно применять типовые решения в рамках, учебной программы учреждения высшего образования по учебной дисциплине;

усвоение основной литературы, рекомендованной учебной программой учреждения высшего образования по учебной дисциплине;

умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;

активная самостоятельная работа на практических, лабораторных занятиях, периодическое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

5 (пять) баллов, зачтено:

достаточные знания в объеме учебной программы учреждения высшего образования по учебной дисциплине;

использование научной терминологии, грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;

владение инструментарием учебной дисциплины, умение его использовать в решении учебных и профессиональных задач;

способность самостоятельно применять типовые решения в рамках учебной программы учреждения высшего образования по учебной дисциплине;

усвоение основной литературы, рекомендованной учебной программой учреждения высшего образования по учебной дисциплине;

умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им сравнительную оценку;

самостоятельная работа на практических, лабораторных занятиях, фрагментарное участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

4 (четыре) балла, зачтено:

достаточный объем знаний в рамках образовательного стандарта высшего образования;

усвоение основной литературы, рекомендованной учебной программой учреждения высшего образования по учебной дисциплине;

использование научной терминологии, логическое изложение ответа на вопросы, умение делать выводы без существенных ошибок;

владение инструментарием учебной дисциплины, умение его использовать в решении стандартных (типовых) задач;

умение под руководством преподавателя решать стандартные (типовые) задачи;

умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им оценку;

работа под руководством преподавателя на практических, лабораторных занятиях, допустимый уровень культуры исполнения заданий.

3 (три) балла, не зачтено:

недостаточно полный объем знаний в рамках образовательного стандарта высшего образования;

знание части основной литературы, рекомендованной учебной программой учреждения высшего образования по учебной дисциплине;

использование научной терминологии, изложение ответа на вопросы с существенными, логическими ошибками;

слабое владение инструментарием учебной дисциплины, некомпетентность в решении стандартных (типовых) задач;

неумение ориентироваться в основных теориях, концепциях и направлениях изучаемой учебной дисциплины;

пассивность на практических и лабораторных занятиях, низкий уровень культуры исполнения заданий.

2 (два) балла, не зачтено:

фрагментарные знания в рамках образовательного стандарта высшего образования;

знания отдельных литературных источников, рекомендованных учебной программой учреждения высшего образования по учебной дисциплине;

неумение использовать научную терминологию учебной дисциплины, наличие в ответе грубых, логических ошибок;

пассивность на практических и лабораторных занятиях, низкий уровень культуры исполнения заданий.

1 (один) балл, не зачтено:

отсутствие знаний и (компетенций) в рамках образовательного стандарта высшего образования, отказ от ответа, неявка на аттестацию без уважительной причины.

11. МЕТОДЫ И СРЕДСТВА РЕАЛИЗАЦИИ СОДЕРЖАНИЯ УЧЕБНОЙ ПРОГРАММЫ И ПОДГОТОВКИ УЧЕБНЫХ, УЧЕБНО-МЕТОДИЧЕСКИХ МАТЕРИАЛОВ

Обучение будет проводиться с использованием классических методов, проектного метода и дистанционных технологий, реализованных на портале SDO.VSTU.BY. Обучающиеся будут обеспечены доступом к порталу SDO.VSTU.BY.

На очных занятиях обучающиеся будут изучать дисциплину непосредственно в компьютерном классе. При проведении занятий будет использоваться следующее программное обеспечение (ПО):

- IDE MS Visual Studio Community,
- веб-браузер.

При подготовке учебных, учебно-методических материалов были использованы следующие средства и технологии:

- Adobe PDF.
- LMS Moodle.

- MS Visual Studio Community 2015.
- Программы для записи и редактирования видео.

12.ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Основная литература

№ п.п.	Авторы	Название учебников и учебных пособий, Год издания
1.	Т. Павловская А.	С/С++. Программирование на языке высокого уровня: учебник для студентов вузов, обучающихся по направлению подготовки дипломированных специалистов "Информатика и вычислительная техника" . - Санкт-Петербург: Питер, 2010
2.	P.S. Deshpande	C & Data Structures = Систруктураданных / P. S. Deshpande, O. G. Kakde. - New Delhi: Dreamtech Press, 2010.
3.	H. Schildt	C++: The Complete Reference = C++: полноуководство / H. Schildt,. - 4th ed. - New Delhi: Tata McGraw Hill Education Private Limited, 2010.
4.	В. В. Лаптев	C++. Объектно-ориентированное программирование: учебное пособие - Санкт-Петербург: Питер, 2008. - 458 с
5.	Б. Керниган,, Д. Ритчи	Язык программирования Си: пер. с англ. /; под ред. В. С. Штармана. - 2-е изд., перераб. и доп. - Москва: Финансы и статистика, 1992.

Дополнительная литература

№ п.п.	Страуструп Б.	Язык программирования C++, спец. изд. – М.,СПб.: “Издательство БИНОМ” – “Невский диалект”, 2001
1.	Дейтел Х, Дейтел П.	Как программировать на С. – М.: ЗАО “Издательство БИНОМ”, 2000.–1008с.
2.	Кнут Д.Э.	Искусство программирования: Учеб. пособие. Т. 3. Сортировка и поиск. – М.: Вильямс, 2000
3.	Д. Э. Кнут	Искусство программирования: пер. с англ. Т. 2 : Получисленные алгоритмы - 3-е изд. - Москва: Вильямс, 2001.
4.	Д. Э. Кнут	Искусство программирования: пер. с англ. Т. 1 : Основные алгоритмы - 3-е изд. - Москва: Вильямс, 2001.