

Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: Котова Лариса Анатольевна
 Должность: Директор филиала
 Дата подписания: 16.03.2023 08:10:46
 Уникальный программный ключ:
 10730ffe6b1ed036b744b6a9d97700b86e5c04a7

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное образовательное учреждение
 высшего образования
 «Национальный исследовательский технологический университет «МИСиС»
 Новотроицкий филиал**

Рабочая программа дисциплины (модуля)

Языки программирования

Закреплена за подразделением Кафедра математики и естествознания (Новотроицкий филиал)

Направление подготовки 09.03.03 Прикладная информатика

Профиль

Квалификация **Бакалавр**
 Форма обучения **очная**
 Общая трудоемкость **10 ЗЕТ**

Часов по учебному плану	360	Формы контроля в семестрах: экзамен 4 зачет 3
в том числе:		
аудиторные занятия	136	
самостоятельная работа	188	
часов на контроль	36	

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	3 (2.1)		4 (2.2)		Итого	
	УП	РП	УП	РП		
Неделя	18		18			
Вид занятий	УП	РП	УП	РП	УП	РП
Лекции	17	17	34	34	51	51
Лабораторные	17	17	17	17	34	34
Практические	17	17	34	34	51	51
В том числе инт.	23	23	29	29	52	52
Итого ауд.	51	51	85	85	136	136
Контактная работа	51	51	85	85	136	136
Сам. работа	93	93	95	95	188	188
Часы на контроль			36	36	36	36
Итого	144	144	216	216	360	360

Программу составил(и):

доцент, Рыбиков Е.В.

Рабочая программа

Языки программирования

Разработана в соответствии с ОС ВО:

Самостоятельно устанавливаемый образовательный стандарт высшего образования Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский технологический университет «МИСиС» по направлению подготовки 09.03.03 Прикладная информатика (уровень бакалавриата) (приказ от 05.03.2020 г. № № 95 о.в.)

Составлена на основании учебного плана:

Направление подготовки 09.03.03 Прикладная информатика Профиль. Прикладная информатика в технических системах, 09.03.03_21_Прикладная информатика_ПрПИВТС_2020.plx , утвержденного Ученым советом ФГАОУ ВО НИТУ "МИСиС" в составе соответствующей ОПОП ВО 21.05.2020, протокол № 10/зг

Утверждена в составе ОПОП ВО:

Направление подготовки 09.03.03 Прикладная информатика Профиль. Прикладная информатика в технических системах, , утвержденной Ученым советом ФГАОУ ВО НИТУ "МИСиС" 21.05.2020, протокол № 10/зг

Рабочая программа одобрена на заседании

Кафедра математики и естествознания (Новотроицкий филиал)

Протокол от 24.06.2021 г., №11

Руководитель подразделения доцент, к.ф.м.н. Гюнтер Д.А.

1. ЦЕЛИ ОСВОЕНИЯ

1.1	Цели освоения дисциплины: формирование компетенций будущих выпускников в области прикладной информатики через ознакомление с общими принципами построения и использования языков программирования, а также развитие навыков проектирования и реализации алгоритмов решения практических задач на языке C/C++.
1.2	Задачи:
1.3	- изучить основные виды языков программирования и принципы алгоритмического подхода от этапа формализации до реализации в виде программного кода;
1.4	- сформировать навыки работы с различными средствами программирования и отладки для создания программного обеспечения на языках высокого уровня.

2. МЕСТО В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Блок ОП:		Б1.Б
2.1	Требования к предварительной подготовке обучающегося:	
2.1.1	Информатика	
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:	
2.2.1	Экономика	
2.2.2	Правоведение	
2.2.3	Основы микропроцессорной техники	
2.2.4	Подготовка к процедуре защиты и защита выпускной квалификационной работы	

3. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ФОРМИРУЕМЫМИ КОМПЕТЕНЦИЯМИ

ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности

Знать:

ОПК-2-33 Работа со строками

1. Строки и класс System.String
2. Операции со строками
3. Форматирование и интерполяция строк
4. Класс StringBuilder

Работа с датами и временем:

1. Структура DateTime

Работа с файловой системой:

1. Работа с дисками
2. Работа с каталогами
3. Работа с файлами. Классы File и FileInfo
4. FileStream. Чтение и запись файла
5. Чтение и запись текстовых файлов. StreamReader и StreamWriter
6. Бинарные файлы. BinaryWriter и BinaryReader

ОПК-2-32 Классы, структуры и пространства имен:

1. Классы и объекты
2. Конструкторы, инициализаторы и деструкторы
3. Класс Program и метод Main. Программы верхнего уровня
4. Структуры
5. Типы значений и ссылочные типы
6. Область видимости (контекст) переменных
7. Модификаторы доступа
8. Свойства
9. Перегрузка методов
10. Статические члены и модификатор static
11. Поля и структуры для чтения

ОПК-2-31 Основы программирования на C#:

1. Структура программы
2. Переменные и константы
3. Литералы
4. Типы данных
5. Консольный ввод-вывод
6. Арифметические операции

7. Поразрядные операции 8. Операции присваивания 9. Преобразования базовых типов данных 10. Условные выражения 11. Конструкция if..else и тернарная операция 12. Циклы 13. Массивы 14. Методы 15. Параметры методов 16. Возвращение значения и оператор return 17. Передача параметров по ссылке и значению. Выходные параметры 18. Массив параметров и ключевое слово params 19. Рекурсивные функции 20. Локальные функции 21. Конструкция switch 22. Перечисления enum

Уметь:

ОПК-2-У3 Для написания клиент-серверного приложения на C# студент должен уметь: 1. Разработать клиентское приложение с помощью ООП 2. Работать с подключениями удаленных (локальных) баз данных 3. Правильно организовать работу с данными таблиц посредством языка SQL 4. Выводить результирующие или итоговые данные в виде отчета
--

ОПК-2-У2 Для решения задач по большим данным и его анализу уметь: 1. Выбрать оптимальную СУБД 2. Разработать реляционную базу данных 3. Работать с первичными и вторичными ключами таблиц 4. Писать запросы с помощью языка SQL 5. Производить манипуляции с данными по средством DML запросов

ОПК-2-У1 Для решения технической задачи студент должен уметь: 1. Определить модель или способ решения задачи 2. Определит входные параметры 3. Определить промежуточные параметры 4. Правильно и грамотно определить ряд типов данных, с которым ему придется работать 5. Определить выходные параметры
--

Владеть:

ОПК-2-В2 Разработка структуры базы данных Firebird Язык SQL для создания структуры БД, связей между ключевыми полями, заполнение данными Подключение к клиентскому приложению и работа с БД

ОПК-2-В1 Основами программирования на Visual Studio C# Создание клиентских и консольных приложений

4. СТРУКТУРА И СОДЕРЖАНИЕ

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Формируемые индикаторы компетенций	Литература и эл. ресурсы	Примечание	КМ	Выполняемые работы
	Раздел 1. Основы алгоритмизации							
1.1	Характеристика предметной области. Классификация задач, обзор парадигм, языков и средств программирования. Характер современного программирования. Понятие и свойства алгоритма. Виды и способы записи алгоритмов. UML как средство проектирования программы. /Лек/	3	6		Л1.1 Л1.2Л2.2 Э1			

1.2	Самостоятельное изучение учебного материала в LMS Canvas: Программная система, как подсистема автоматизированной системы. Основные термины и определения. Понятие жизненного цикла программного обеспечения. Методологии разработки программного обеспечения. /Ср/	3	24		Л1.1 Л1.2Л2.2 Э1			
1.3	Введение в языки программирования. Общие принципы построения и использования языков программирования. Знакомство со средой программирования /Лаб/	3	4		Л1.1 Л1.2Л2.2 Л2.3Л3.2 Л3.3 Э1			
1.4	Изучение основных элементов блок-схем для представления алгоритмов /Пр/	3	4		Л1.1 Л1.2Л2.2 Л2.3 Э1			
Раздел 2. Язык программирования Си								
2.1	Разработка линейной программы /Лаб/	3	6		Л1.1 Л1.2Л2.2 Л2.3Л3.2 Л3.3 Э1			
2.2	Построение линейного и ветвящегося алгоритма. Выдача заданий для выполнения контрольной работы №1. /Пр/	3	6		Л1.1 Л1.2Л2.2 Л2.3Л3.1 Э1			
2.3	Общая характеристика языка: место в классификации языков, синтаксис и семантика языка, основные объекты языка, операторы, структура программы. Встроенные типы языка С. Неявное и явное приведение (преобразование) встроенных типов. /Лек/	3	5		Л1.1 Л1.2Л2.1 Л2.2 Э1			
2.4	Самостоятельное изучение учебного материала в LMS Canvas: Фон-неймановская архитектура. Виртуальная машина языка программирования. Память. Среда вычислений. Переменная. Выражение. Команда. Составные команды. Подпрограммы. Модули. /Ср/	3	24		Л1.1 Л1.2Л2.1 Л2.2 Э1			
Раздел 3. Конструкции структурного программирования в Си								

3.1	Алгоритмические конструкции ветвления: полная и неполная форма, блок-схемы. Условная операция. Условный оператор и оператор выбора в языке Си: структура оператора, полная и неполная формы, использование сложных условий. /Лек/	3	6		Л1.1 Л1.2Л2.1 Л2.2 Э1			
3.2	Самостоятельное изучение учебного материала в LMS Canvas: Циклические алгоритмы: понятие, виды. Алгоритмическая конструкция цикла с предусловием, с постусловием, цикл с параметром (понятие, использование, блок-схема). Выполнение контрольной работы №1. Подготовка к зачету. /Ср/	3	45		Л1.1 Л1.2Л2.1 Л2.2Л3.1 Э1			
3.3	Построение циклического алгоритма /Пр/	3	7		Л1.1 Л1.2Л2.1 Л2.2 Л2.3 Э1			
3.4	Разработка циклической программы /Лаб/	3	7		Л1.1 Л1.2Л2.1 Л2.2 Л2.3Л3.2 Л3.3 Э1			
Раздел 4. Структурированные типы данных в Си								
4.1	Линейный массив: понятие массива, объявление, инициализация массива, индексация элементов. Формирование и вывод массива. Типовые алгоритмы для работы с линейными массивами. Алгоритмы сортировки линейного массива. Нелинейные структуры данных. Графы и деревья. Определение, способы реализации. /Лек/	4	16		Л1.1 Л1.2Л2.1 Л2.2 Э1			

4.2	Самостоятельное изучение учебного материала в LMS Canvas: Строка и структура в языке Си: понятие, описание и использование. Простые сортировки. Сортировки Шелла, быстрая, слиянием. Специальные сортировки: подсчетом, поразрядная, карманная. Линейный поиск, бинарный поиск. Поиск подстроки в строке. Алгоритмы Кнута-Морриса-Пратта, Бойера-Мура, Рабина-Карпа. Бинарные деревья: определение, способы построения. Сбалансированные деревья: бинарное, красное-черное. /Ср/	4	29		Л1.1 Л1.2Л2.1 Л2.2 Э1			
4.3	Построение алгоритма обработки массива /Пр/	4	8		Л1.1 Л1.2Л2.1 Л2.2 Л2.3 Э1			
4.4	Построение алгоритма обработки строк /Пр/	4	8		Л1.1 Л1.2Л2.1 Л2.2 Л2.3 Э1			
4.5	Обработка массивов в Си /Лаб/	4	4		Л1.1 Л1.2Л2.1 Л2.2 Л2.3Л3.2 Л3.3 Э1			
4.6	Обработка строк в Си /Лаб/	4	4		Л1.1 Л1.2Л2.1 Л2.2 Л2.3Л3.2 Л3.3 Э1			
Раздел 5. Функции в Си								
5.1	Понятие, объявление, прототипы функций. Вызов функции. Локальные и глобальные переменные. Передача данных в функции. /Лек/	4	12		Л1.1 Л1.2Л2.2 Э1			
5.2	Построение алгоритма программы, использующей функции. Выдача заданий для контрольной работы №2. /Пр/	4	10		Л1.1 Л1.2Л2.2 Л2.3Л3.1 Э1			
5.3	Функции в Си /Лаб/	4	6		Л1.1 Л1.2Л2.2 Л2.3Л3.2 Л3.3 Э1			
5.4	Самостоятельное изучение учебного материала в LMS Canvas: Определение и использование функций. Передача параметров. Формальные и фактические параметры. /Ср/	4	18		Л1.1 Л1.2Л2.2 Э1			
Раздел 6. Файлы в Си								

6.1	Файловый ввод-вывод в стиле. Работа с бинарными файлами. Структуры и бинарные файлы. /Лек/	4	6		Л1.1 Л1.2Л2.2 Э1			
6.2	Построение алгоритма программ для работы с файлами. /Пр/	4	8		Л1.1 Л1.2Л2.2 Л2.3 Э1			
6.3	Структуры и файлы в Си /Лаб/	4	3		Л1.1 Л1.2Л2.2 Л2.3Л3.2 Л3.3 Э1			
6.4	Самостоятельное изучение учебного материала в LMS Canvas: Текстовый файл. Бинарные файлы. Форматный обмен с файлами. Выполнение контрольной работы №2. Подготовка к экзамену. /Ср/	4	48		Л1.1 Л1.2Л2.2Л3.1 Э1			
6.5	Проведение экзамена /Экзамен/	4	36					

5. ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ

5.1. Контрольные мероприятия (контрольная работа, тест, коллоквиум, экзамен и т.п), вопросы для самостоятельной подготовки

Код КМ	Контрольное мероприятие	Проверяемые индикаторы компетенций	Вопросы для подготовки
КМ1	Контрольная работа "Основы программирования на С#"	ОПК-2-31	<ol style="list-style-type: none"> 1. Арифметические операции на языке С#. 2. Преобразования типов. 3. Логические операции и операции отношения. 4. Операции инкремента и декремента. 5. Объявление и использование символических констант. 6. Операция выбора. 7. Оператор выбора if 8. Оператор цикла do .. while 9. Оператор цикла foreach 10. Простейшие приёмы работы с одномерными массивами

КМ2	Контрольная работа "Классы и объекты в С#"	ОПК-2-32	<p>Описать объект в виде класса, класс должен содержать конструкторы, константы, поля, методы, свойства, перегрузки, а также применить к полям различные модификаторы доступа. Показать создание экземпляра класса и обращение к его основным элементам, вывод информации класса (поля, свойства и т.д.), кроме этого показать доступ к методам по его имени (без создания экземпляра класса). Максимально показать все те элементы класса, которые были рассмотрены на практических занятиях. Выбрать вариант по порядковому номеру из журнала и описать следующий абстрактный класс:</p> <ol style="list-style-type: none"> 1. Класс «Human» (Описание класса – человек). 2. Класс «Auto» (Описание класса - автомобиль). 3. Класс «Magazine» (Описание класса – журнал, газета). 4. Класс «Football team» (Описание класса – футбольная команда). 5. Класс «Weather» (Описание класса - погода). 6. Класс «Shoes» (Описание класса - обувь). 7. Класс «Music» (Описание класса - музыка). 8. Класс «Movie» (Описание класса - кино). 9. Класс «Rectangle» (Описание класса – фигура Прямоугольника). 10. Класс «Circle» (Описание класса – фигуры окружность). 11. Класс «Computer» (Описание класса - компьютер). 12. Класс «Building» (Описание класса – строение, здание) 13. Класс «Bank» (Описание класса - банк). 14. Класс «Bookmaker» (Описание класса – букмекерская компания).
-----	--	----------	---

5.2. Перечень работ, выполняемых по дисциплине (Курсовая работа, Курсовой проект, РГР, Реферат, ЛР, ПР и т.п.)

Код работы	Название работы	Проверяемые индикаторы компетенций	Содержание работы
Р1	Лабораторная работа № 1 Изучение среды разработки VS	ОПК-2-31	<p>Цель лабораторной работы: изучить среду быстрой разработки приложений Visual Studio. Научится размещать и настраивать внешний вид элементов управления на форме.</p> <p>Темы:</p> <ol style="list-style-type: none"> 1.1 Интегрированная среда разработчика Visual Studio 1.2. Настройка формы 1.3. Размещение элементов управления на форме 1.4. Размещение строки ввода 1.5. Размещение надписей 1.6. Написание программы обработки события 1.7. Написание программы обработки события нажатия кнопки 1.8. Написание программы обработки события загрузки формы 1.9. Запуск и работа с программой 1.10. Динамическое изменение свойств 1.11. Выполнение индивидуального задания
Р2	Лабораторная работа № 2 Линейные алгоритмы	ОПК-2-31	<p>Цель лабораторной работы: научиться составлять каркас простейшей программы в среде Visual Studio. Написать и отладить программу линейного алгоритма.</p> <p>Темы:</p> <ol style="list-style-type: none"> 2.1. Структура приложения 2.2. Работа с проектом 2.3. Описание данных 2.4. Ввод/вывод данных в программу 2.5. Арифметические действия и стандартные функции 2.6. Пример написания программы 2.7. Выполнение индивидуального задания

P3	Лабораторная работа № 3 Разветвляющиеся алгоритмы	ОПК-2-31	<p>Цель лабораторной работы: научиться пользоваться элементами управления для организации переключений (RadioButton). Написать и отладить программу разветвляющегося алгоритма.</p> <p>Темы: 3.1. Логические переменные и операции над ними 3.2. Условные операторы 3.3. Кнопки-переключатели 3.4. Пример написания программы 3.4.1. Создание формы 3.4.2. Создание обработчиков событий</p>
P4	Лабораторная работа №4. Циклические алгоритмы	ОПК-2-31	<p>Цель лабораторной работы: изучить простейшие средства отладки программ в среде Visual Studio. Составить и отладить программу циклического алгоритма.</p> <p>Темы: 4.1. Операторы организации циклов 4.2. Цикл с предусловием 4.3. Цикл с постусловием 4.4. Цикл с параметром 4.5. Средства отладки программ 4.6. Порядок выполнения задания</p>
P5	Лабораторная работа №5. Классы и объекты	ОПК-2-32	<p>Цель лабораторной работы: изучить основные понятия, относящиеся к классам и объектам, освоить динамическое создание объектов в программном коде.</p> <p>Темы: 5.1 Классы и объекты 5.2 Динамическое создание объектов 5.3 Область видимости 5.4 Операции is и as 5.5 Сведения, передаваемые в событие</p>
P6	Лабораторная работа №6. Работа со строками	ОПК-2-31	<p>Цель лабораторной работы: изучить правила работы с элементом управления ListBox. Написать программу для работы со строками.</p> <p>Темы: 6.1. Строковый тип данных 6.2. Более эффективная работа со строками 6.3. Элемент управления ListBox 6.3. Порядок выполнения индивидуального задания</p>
P7	Лабораторная работа №7. Одномерные массивы	ОПК-2-31	<p>Цель лабораторной работы: Изучить способы получения случайных чисел. Написать программу для работы с одномерными массивами.</p> <p>Темы: 7.1. Работа с массивами 7.2. Случайные числа 7.3. Порядок выполнения индивидуального задания</p>
P8	Лабораторная работа №8. Многомерные массивы	ОПК-2-31	<p>Цель лабораторной работы: изучить свойства элемента управления DataGridView. Написать программу с использованием двухмерных массивов.</p> <p>Темы: 8.1. Двухмерные массивы 8.2. Элемент управления DataGridView 8.3. Порядок выполнения задания</p>

P9	Лабораторная работы № 1 II семестр	ОПК-2-У2	<p>Разработать структуру базы данных в виде таблиц и их описания (с типами данных полей)</p> <p>Вариант 1 «Учёт работ строительной компании» Вариант 2 «Учёт материалов на складе» Вариант 3 «Учёт студентов, проживающих в общежитиях» Вариант 4 «Учёт работы поликлиники» Вариант 5 «Учёт отпусков сотрудников» Вариант 6 «Учёт арендуемых помещений» Вариант 7 «Учёт работы автомастерской» Вариант 8 «Учёт работы туристического агентства» Вариант 9 «Учёт библиотечного фонда» Вариант 10 «Учёт материальных ценностей» Вариант 11 «Учёт выступлений студентов на научных конференциях» Вариант 12 «Учёт работы научных конференций» Вариант 13 «Учёт успеваемости студентов» Вариант 14 «Учёт посещаемости студентов» Вариант 15 «Расписания движения поездов» Вариант 16 «Учет домашних животных» Вариант 17 «Учет банковских операций» Вариант 18 «Учет матчей» Вариант 19 «Поликлиника» Вариант 20 «Регистрация автомобилей»</p>
----	---------------------------------------	----------	---

5.3. Оценочные материалы, используемые для экзамена (описание билетов, тестов и т.п.)

В 3 семестре экзамен по дисциплине не предусмотрен.

Формой промежуточной аттестации по дисциплине является зачет. Зачет ставится при выполнении всех проверочных заданий по итогам каждого раздела более, чем на 50%.

В 4 семестре формой промежуточной аттестации по дисциплине является экзамен.

Ниже представлен образец билета для экзамена, проводимого в устной форме.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
 Федеральное государственное автономное образовательное учреждение высшего образования
 НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ
 «МИСиС»
 НОВОТРОИЦКИЙ ФИЛИАЛ

Кафедра электроэнергетики и электротехники

БИЛЕТ К ЭКЗАМЕНУ № 0

Дисциплина: «Языки программирования»
 Направление: 09.03.03 «Прикладная информатика»
 Форма обучения: очная

Форма проведения экзамена: устная

1. Быстрая сортировка. Обоснование корректности алгоритма.
2. Функции в C++. Объявление и определение функций.

Составил доцент: _____ Р.Е. Мажирина

Зав. кафедрой ЭиЭ: _____ Р.Е. Мажирина

«01» сентября 2020 г.

Дистанционно экзамен проводится в LMS Canvas. Экзаменационный тест содержит 30 заданий. На решение отводится 30 минут. Разрешенные попытки - две. Зачитывается наилучший результат.

Образец заданий для экзамена, проводимого дистанционно в LMS Canvas:

1. Этап проектирования задачи на ЭВМ, на котором анализируется условие задачи, определяются исходные данные и результаты, устанавливается зависимость между величинами, рассматриваемыми в задаче, называется ...
 - а) постановка задачи
 - б) разработка алгоритма
 - в) программирование

- г) тестирование и отладка
2. Этап проектирования задачи на ЭВМ, на котором происходит запуск тестовой программы с использованием контрольных примеров, называется ...
- построение математической модели
 - разработка алгоритма
 - программирование
 - тестирование и отладка
3. Этап проектирования задачи на ЭВМ, на котором происходит формулировка условия задачи, называется ...
- постановка задачи
 - построение математической модели
 - разработка алгоритма
 - программирование
4. Этап проектирования задачи на ЭВМ, на котором программа записывается на компьютерном языке и вводится в ЭВМ, называется ...
- постановка задачи
 - построение математической модели
 - разработка алгоритма
 - программирование
5. На этапе тестирования и отладки при проектировании задачи на ЭВМ происходит ...
- получение результата
 - обнаружение ошибок
 - запись алгоритма на языке программирования
 - формализация задачи
6. Алгоритмом является следующее описание
- Пойди туда, не знаю куда. Принеси то, не знаю что.
 - Возьми, что нужно. Сделай как следует. Получишь то, что желаешь.
 - Сделай шаг вперед. Сделай шаг назад. Начни сначала.
 - 0010101101 101001.
7. Исполнитель алгоритма – это ...
- человек или компьютер, умеющий выполнять определённый набор действий
 - понятное и точное предписание необходимых действий
 - определённые условия выполнения действий
 - элемент, связывающий этапы выполнения алгоритма
8. Графическое задание алгоритма (блок-схемы) – это ...
- способ представления алгоритма с помощью геометрических фигур
 - система обозначения правил для единообразной и точной записи алгоритмов их исполнения
 - схематичное изображение в произвольной форме
 - формализованная задача
9. Линейный алгоритм – это ...
- способ представления алгоритма с помощью геометрических фигур
 - набор команд, которые выполняются последовательно друг за другом
 - понятное и точное предписание исполнителю для выполнения различных ветвлений
 - последовательное выполнение команд
10. Циклический алгоритм используется при вычислении ...
- суммы всех чисел от 1 до 100
 - площади трапеции
 - корня квадратного уравнения
 - суммы двух чисел, введенных с клавиатуры
11. Разветвляющийся алгоритм – это ...
- присутствие в алгоритме хотя бы одного условия
 - набор команд, которые выполняются последовательно друг за другом
 - многократное выполнение одних и тех же действий
 - алгоритм, использующий подпрограммы
12. Что такое язык программирования?
- система, определяющая воздействие одного объекта на другой
 - формальная знаковая система, предназначенная для описания алгоритма в форме
 - выделение существенных характеристик некоторого объекта

13. Что определяет язык программирования?
а) организацию соответствующих условий
б) последовательность двоичных кодов
в) набор лексических, синтаксических и семантических правил
14. На какие группы можно разделить множество языков программирования?
а) язык ассемблера
б) языки низкого и высокого уровней
в) язык мнемокодов
15. Язык Ассемблера – это ...
а) машинно-зависимый язык низкого уровня, в котором короткие мнемонические имена соответствуют отдельным машинным командам
б) система для разработки новых программ на конкретном языке программирования
в) машинный язык, который понимает процессор
г) язык программирования для комбинирования компонентов, набор которых создается заранее при помощи других языков
16. Что составляется из символов?
а) множество символов, используемых на языке
б) ключевые слова языка
в) двоичные коды
17. Что называется трансляцией?
а) процесс обработки программы
б) процесс интерпретации
в) процесс перевода.
18. Система программирования – это ...
а) машинно-зависимый язык низкого уровня, в котором короткие мнемонические имена соответствуют отдельным машинным командам
б) система для разработки новых программ на конкретном языке программирования
в) машинный язык, который понимает процессор
г) язык программирования для комбинирования компонентов, набор которых создается заранее при помощи других языков
19. Ошибка, которая не обнаруживаются транслятором: ...
а) отсутствие описания переменных
б) неверное написание служебных слов
в) деление на 0
г) бесконечный цикл/неправильное условие окончания цикла
20. Ошибка, которая обнаруживаются транслятором: ...
а) неверное написание служебных слов
б) несогласованность скобок
в) неверное определение порядка арифметических действий
г) неполный учёт возможных условий
21. Файлы с текстами программ на языке C++ имеют расширение
а) *.h, *.hpp, *.c или *.cpp
б) *.txt или *.doc
в) *.obj или *.lib
22. Заголовочные файлы (с расширением *.h или *.hpp) в языке C++ используются для
а) объявления в них переменных программы
б) раздельной компиляции модулей программы
в) хранения массивов данных программы
23. Заголовочные файлы (с расширением *.h или *.hpp) в языке C++ подключаются к компилируемому файлу
а) с помощью директивы #include
б) с помощью директивы #inpute
в) с помощью директивы #insert
24. Точкой входа в программу на языке C++ (из перечисленных) является функция
а) begin()
б) start()
в) main()
25. Один и тот же заголовочный файл (с расширением *.h или *.hpp) можно подключать

- а) только к одному модулю программы
б) только к двум модулям программ
в) к любому количеству модулей программы
26. Программа на языке C++ начинает свою работу
а) с первой строки первого модуля программы
б) с функции main() или WinMain()
в) с произвольного места, помеченного программистом директивой #begin
27. Интегрированная среда разработчика C++ Builder позволяет создавать
а) только консольные приложения
б) только приложения с визуальным интерфейсом
в) и консольные и визуальные приложения –по выбору программиста.
28. С точки зрения языка C++ выражение является истинным, если
а) это выражение равно 0
б) это выражение не равно 0
в) это выражение равно 1
29. С точки зрения языка C++ выражение является ложным, если
а) это выражение равно 0
б) это выражение не равно 0
в) это выражение равно 1
30. В результате выполнения программы
int x, y;
x = 0;
y = 0;
if (x)
{
y = 1;
}
переменная y получит значение
а) 0
б) 1
в) -1
31. Результат выражения $27 \text{ div } 2$ равен ...
а) 13
б) 54
в) 25
г) 29
32. Результат выражения $27 \text{ mod } 4$ равен ...
а) 108
б) 3
в) 7
г) 23
33. Основное отличие операторов прерывания циклов break и continue состоит в том, что
а) оператор break прерывает выполнение содержащего его цикла, оператор continue только текущей итерации содержащего его цикла
б) оператор continue прерывает выполнение содержащего его цикла, оператор break только текущей итерации содержащего его цикла
в) оператор break может использоваться в циклах for, оператор continue не может
34. Основным типом подпрограмм в языке C++ является
а) процедура
б) функции
в) оператор повторений
35. Если в функции на языке C++ отсутствует оператор return, то такая функция
а) не будет возвращать значения
б) будет возвращать значение 1
в) будет возвращать значение 0

5.4. Методика оценки освоения дисциплины (модуля, практики. НИР)

Критерии оценки ответов на экзамене, проводимом в устной форме

Оценка «Отлично» ставится, если

- на теоретические вопросы даны развернутые ответы, при необходимости изложен математический аппарат (формулы, графики и т.д.) приведены соответствующие схемы, таблицы, рисунки и т.д., правильно решена задача
- обучающийся хорошо ориентируется в материале, владеет терминологией, приводит примеры, обосновывает, анализирует, высказывает свою точку зрения по анализируемым явлениям и процессам, правильно применяет полученные знания при решении практических задач. Ответы излагаются свободно, уверенно без использования листа устного опроса

Оценка «Хорошо» ставится, если

- на теоретические вопросы даны полные ответы, но имела место неточность в определении каких-либо понятий, явлений и т.д. Задача решена.
- обучающийся ориентируется в материале хорошо, но допускает ошибки при формулировке, описании отдельных категорий

Оценка «Удовлетворительно» ставится, если

- на теоретические вопросы даны общие неполные ответы
- обучающийся слабо ориентируется в материале, не может решать задачи, не может привести пример, не может анализировать и обосновывать

Оценка «Неудовлетворительно» ставится, если

- не решена задача и правильный ответ дан на один вопрос (либо ни на один)
- обучающийся в материале дисциплины практически не ориентируется, т.е. не может дать даже общих сведений по вопросу.

Критерии оценки ответов на экзамене, проводимом в дистанционной форме в LMS Canvas

$90 \leq$ Процент верных ответов ≤ 100 - отлично

$75 \leq$ Процент верных ответов < 90 - хорошо

$60 \leq$ Процент верных ответов < 75 – удовлетворительно

Критерии оценки выполнения расчетно-графической работы:

1. Теоретические сведения изложены в достаточном объеме, четко и последовательно
2. Исследуются и сравниваются разные подходы, методики, приводятся собственные суждения и выводы
3. Расчетные задачи решены верно, в полном объеме, приведены логические пояснения
4. Расставлены ссылки на источники
5. Текст написан грамотно, стилистически выдержан
6. Текст оформлен в соответствии с требованиями

Работа оценивается на отлично, если:

- теоретические сведения изложены в достаточном объеме, четко и последовательно, использованы выводы (позиции, мнения и др.) известных ученых, профессионалов, исследуются и сравниваются разные подходы, методики, приводятся собственные суждения и выводы, имеются примеры, даются ссылки на источники, текст написан грамотно, стилистически выдержан и оформлен в соответствии с требованиями.
- расчетные задачи решены в полном объеме, приведены пояснения.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение работы оценивается как хорошее, если она соответствует всем критериям, перечисленным выше, но отсутствует описание и сравнения разных подходов, методик и т.д. с последующим формированием собственных выводов на данный счет. Расчетные задачи решены полностью, не приведены этапы решения, отсутствуют пояснения.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение работы оценивается как удовлетворительное, если она соответствует всем критериям, перечисленным выше, но в первой главе работы отсутствуют описание и сравнения разных подходов, методик и т.д. с последующим формированием собственных выводов на данный счет. Расчетные задачи решены не полностью, отсутствуют пояснения и этапы решения.

Если расчетно-графическая работа не соответствует критериям перечисленным выше, то оценивается неудовлетворительно.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л1.1	Плясунов Д.Ю.	Программирование на VISUAL BASIC: Учеб. пособие		Новотроицк, 2002,

	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л1.2	Литвиненко В.А.	Программирование на C++ задач на графах: учебное пособие		Таганрог : Издательство Южного федерального университета, 2016, http://biblioclub.ru/index.php?page=book&id=493220
6.1.2. Дополнительная литература				
	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л2.1	Березин Б.И.	Начальный курс С и С++ : учебное пособие: учебное пособие		Москва : Диалог-МИФИ, 2012, http://biblioclub.ru/index.php?page=book&id=448000
Л2.2	Павловская Т.А.	С/С++.Программирование на языке высокого уровня: учебник		СПб.: Питер, 2012,
Л2.3	Павловская Т.А.	С/С++. Структурное и объектно-ориентированное программирование. Практикум: учеб.пособие		СПб.: Питер, 2010,
6.1.3. Методические разработки				
	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л3.1	Плясунов Д.Ю.	Технология программирования : Метод. указания		Новотроицк, 2002,
Л3.2	Быковец Н.П.	Алгоритмические языки высокого уровня: Методические указания для выполнения лабораторных работ		Оренбург : ИПК ГОУ ОГУ, 2004,
Л3.3	Малиновская Е.А., Рыскаленко Р.А.	Языки программирования: лабораторный практикум, Ч. 1		Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2016, http://biblioclub.ru/index.php?page=book&id=467412
6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»				
Э1	https://lms.misis.ru/enroll/NMKMRA - курс Языки программирования на платформе LMS Canvas		https://lms.misis.ru/enroll/NMKMRA	
6.3 Перечень программного обеспечения				
6.4. Перечень информационных справочных систем и профессиональных баз данных				
И.1	http://www.intuit.ru/ - Каталог курсов Интернет Университета Информационных Технологий			
И.2	http://www.compress.ru – Журнал «КомпьютерПресс»			
И.3	http://www.osp.ru – Издательство «Открытые системы»			
И.4	http://www.cnews.ru – Издание о высоких технологиях			
И.5	http://www.it-daily.ru – Новости российского ИТ-рынка			

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

Освоение дисциплины предполагает как проведение традиционных аудиторных занятий, так и работу в электронной информационно-образовательной среде НИТУ «МИСиС» (ЭИОС), частью которой непосредственно предназначенной для осуществления образовательного процесса является Электронный образовательный ресурс LMS Canvas. Он доступен по URL адресу <https://lms.misis.ru> и позволяет использовать специальный контент и элементы электронного обучения и дистанционных образовательных технологий. LMS Canvas используется преимущественно для асинхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет».

Чтобы эффективно использовать возможности LMS Canvas, а соответственно и успешно освоить дисциплину, нужно:

- 1) зарегистрироваться на курс. Для этого нужно перейти по ссылке ... Логин и пароль совпадает с логином и паролем от личного кабинета НИТУ МИСиС;
- 2) в рубрике «В начало» ознакомиться с содержанием курса, вопросами для самостоятельной подготовки, условиями допуска к аттестации, формой промежуточной аттестации (зачет/экзамен), критериями оценивания и др.;
- 3) в рубрике «Модули», заходя в соответствующие разделы изучать учебные материалы, размещенные преподавателем. В т.ч. пользоваться литературой, рекомендованной преподавателем, переходя по ссылкам;
- 4) в рубрике «Библиотека» возможно подбирать для выполнения письменных работ (контрольные, домашние работы, курсовые работы/проекты) литературу, размещенную в ЭБС НИТУ «МИСиС»;
- 5) в рубрике «Задания» нужно ознакомиться с содержанием задания к письменной работе, сроками сдачи, критериями оценки. В установленные сроки выполнить работу(ы), подгрузить здесь же для проверки. Удобно называть файл работы следующим образом (название предмета (сокращенно), группа, ФИО, дата актуализации (при повторном размещении)).

Например, Экономика Иванов И.И. БМТ-19_20.04.2020. Если работа содержит рисунки, формулы, то с целью сохранения форматирования ее нужно подгружать в pdf формате.

Работа, подгружаемая для проверки, должна:

- содержать все структурные элементы: титульный лист, введение, основную часть, заключение, список источников, приложения (при необходимости);
- быть оформлена в соответствии с требованиями.

Преподаватель в течение установленного срока (не более десяти дней) проверяет работу и размещает в комментариях к заданию рецензию. В ней он указывает как положительные стороны работы, так замечания. При наличии в рецензии замечаний и рекомендаций, нужно внести поправки в работу, подгрузить ее заново для повторной проверки. При этом важно следить за сроками, в течение которых должно быть выполнено задание. При нарушении сроков, указанных преподавателем возможность подгрузить работу остается, но система выводит сообщение о нарушении сроков. По окончании семестра подгрузить работу не получится;

6) в рубрике «Тесты» пройти тестовые задания, освоив соответствующий материал, размещенный в рубрике «Модули»;

7) в рубрике «Оценки» отслеживать свою успеваемость;

8) в рубрике «Объявления» читать объявления, размещаемые преподавателем, давать обратную связь;

9) в рубрике «Обсуждения» создавать обсуждения и участвовать в них (обсуждаются общие моменты, вызывающие вопросы у большинства группы). Данная рубрика также может быть использована для взаимной проверки;

10) проявлять регулярную активность на курсе.

Преимущественно для синхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет» используется Microsoft Teams (MS Teams). Чтобы полноценно использовать его возможности нужно установить приложение MS Teams на персональный компьютер и телефон. Старостам нужно создать группу в MS Teams.

Участие в группе позволяет:

- слушать лекции;
- работать на практических занятиях;
- быть на связи с преподавателем, задавая ему вопросы или отвечая на его вопросы в общем чате группы в рабочее время с 9.00 до 17.00;
- осуществлять совместную работу над документами (вкладка «Файлы»).

При проведении занятий в дистанционном синхронном формате нужно всегда работать с включенной камерой.

Исключение – если преподаватель попросит отключить камеры и микрофоны в связи с большими помехами. На аватарках должны быть исключительно деловые фото.

При проведении лекционно-практических занятий ведется запись. Это дает возможность просмотра занятия в случае невозможности присутствия на нем или при необходимости вновь обратиться к материалу и заново его просмотреть.