**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ**

**Филиал федерального государственного бюджетного образовательного учреждения высшего образования «Мурманский арктический государственный университет»**

**в г. Апатиты**

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**Б1.Б.5 Информатика**

(шифр дисциплины и название в строгом соответствии

с федеральным государственным образовательным стандартом и учебным планом)

**образовательной программы   
по направлению подготовки бакалавриата**

**09.03.02 «Информационные системы и технологии»**

**профиль: Общий профиль**

(код и наименование направления подготовки

с указанием профиля (наименования магистерской программы)

**очная форма обучения**

форма обучения

|  |  |
| --- | --- |
| **Составитель:**  Тоичкин Н.А., к.т.н.,  доцент кафедры  информатики, вычислительной техники и информационной безопасности | Утверждено на заседании кафедры информатики, вычислительной техники и информационной безопасности  (протокол № \_\_ от \_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ 2016г.)  Зав. кафедрой  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись Ф.И.О. |

**Структура рабочей программы дисциплины**

**1. НАИМЕНОВАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ). Б1.Б.5 Информатика**

**2. АННОТАЦИЯ К ДИСЦИПЛИНЕ**

Цель дисциплины: воспитание у студентов информационной культуры; обучение теоретическим основам и практическим навыкам работы с аппаратным и программным обеспечением компьютера.

В результате изучения дисциплины студент должен:

***знать:***

* основы построения и архитектуры ЭВМ;
* принципы обработки информации на ЭВМ;
* основные понятия и терминологию в области вычислительной техники;
* технические и эксплуатационные характеристики компьютеров;
* современное состояние и тенденции развития ЭВМ;
* основы современных информационных технологий и систем;
* процессы сбора, передачи, обработки и накопления информации;
* задачи профессиональной предметной области, решаемые на персональных компьютерах;
* технические и программные средства реализации информационных процессов;
* принципы работы сетевого обеспечения;
* принципы разработки программных приложений;
* принципы работы систем управления базами данных.

***уметь:***

* работать с электронными таблицами и базами данных;
* разрабатывать блок схемы алгоритмов;
* работать с математическими программными пакетами (MathCAD);
* выполнять алгоритмическую постановку задачи;
* выбирать базовую конфигурацию компьютера.

***владеть:***

* навыками работы в операционной системе Windows;
* навыками работы в электронных таблицах;
* навыками работы с системами управления базами данных;
* навыками алгоритмизации;
* навыками разработки программ в современных средах разработки приложений.

**3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ** **РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ**

В результате освоении содержания дисциплины «Информатика» обучающийся должен обладать следующими компетенциями:

* владением широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий (ОПК-1);
* способностью использовать современные компьютерные технологии поиска информации для решения поставленной задачи, критического анализа этой информации и обоснования принятых идей и подходов к решению (ОПК-5);

**4. УКАЗАНИЕ МЕСТА ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ**

Дисциплина входит в вариативную часть профессионального цикла образовательной программы бакалавра по направлению «Информационные системы и технологии».

Для освоения данной дисциплины обучающиеся используют знания, умения, навыки, способы деятельности и установки, которые они получили в процессе изучения дисциплин средней общеобразовательной школы: «Математика», «Информатика».

В свою очередь, «Информатика» представляет собой методологическую базу для дисциплин: «Программирование», «Технологии программирования», «Архитектура информационных систем».

**5. 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 дисциплины составляет 4 зачетных единицы или 144 часа.

(из расчета 1 ЗЕТ= 36 часов).

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Курс | Семестр | Трудоемкость в ЗЭТ | Общая трудоемкость (час) | Контактная работа | | | Всего контактных часов | Из них в интерактивных формах | Кол-во часов на СРС | Форма  контроля |
| ЛК | ПР | ЛБ |
| **1** | **12** | **7** | **252** | **32** | **32** | **32** | **96** | **17** | **156**  **(из них 72 ч для подготовки к экзамену**  **36 ч - 1 сем.**  **36 ч. - 2 сем.)** | **экзамен** |

**6. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ (РАЗДЕЛАМ) С УКАЗАНИЕМ ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ИЛИ АСТРОНОМИЧЕСКИХ ЧАСОВ И ВИДОВ УЧЕБНЫХ ЗАНЯТИЙ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Наименование  раздела, темы | Контактная работа | | | Всего контактных часов | Из них в интерактивной форме | Кол-во часов на  СРС |
| ЛК | ПР | ЛБ |
| 1 | **Введение в информатику.**  Информатика как наука. *Понятие информации.* Место и роль информации в современном обществе. Структура и задачи информатики. Основные определения информатики. Информационные технологии и информационные системы. | 2 | - | - | 2 | - | 2 |
| 2 | **Понятие информации и ее свойства.**  Информация и ее адекватность. Меры информации. Общая характеристика процессов сбора, передачи, обработки и накопления информации. Качество информации. Системы классификации информации. Кодирование информации в информационных системах. Общая характеристика процессов сбора, передачи, обработки и накопления информации. Информационный процесс в автоматизированных системах. Информационный ресурс и его составляющие. Информационные технологии. | 2 | - | - | 2 | - | 2 |
| 3 | **Системы счисления.**  Позиционные системы счисления. Непозиционные системы счисления. Методы перевода чисел. Форматы представления чисел с плавающей запятой. Двоичная арифметика. Смешанные системы счисления. Системы счисления и архитектура компьютеров. Выполнение арифметических операций над целыми числами. | 2 | 2 | - | 4 | 2 | 4 |
| 4 | **Единицы представления информации в памяти ЭВМ.**  Коды: прямой, обратный, дополнительный, модифицированный. Выполнение арифметических операций над числами с фиксированной и плавающей запятой. Информационные основы контроля работы цифровых автоматов. Систематические коды. Контроль по четности, нечетности, по Хеммингу. Представление символьной и графической информации в ЭВМ.  Сжатие данных. Алгоритмы сжатия. | 2 | 2 | - | 4 | 2 | 4 |
| 5 | **Введение в алгебру логики.**  Понятие высказывания. Основные законы и постулаты алгебры логики. Логические операции. Таблицы истинности. Логические формулы. Законы алгебры логики. Методы решения логических задач. Представление функций алгебры логики. Логический синтез переключательных и вычислительных схем. Построение логических схем в программе MMLogic. | 2 | - | 4 | 6 | 2 | 4 |
| 6 | **Основы элементной базы цифровых автоматов.**  Комбинационные схемы и цифровые автоматы. Логические элементы. Схемотехника логических элементов. Элементы интегральных схем. Изучение алгоритма работы цифровых микросхем и проверка их работоспособности. | 2 | - | - | 2 | - | 4 |
| 7 | **Элементы теории алгоритмов.**  Модели решения функциональных и вычислительных задач алгоритмизация и программирование; языки программирования высокого уровня. Понятие и свойства алгоритма. Принцип программного управления. Машины Тьюринга. | 2 | - | - | 2 | - | 4 |
| 8 | **Основы алгоритмизации.**  Алгоритмизация и программирование. Понятие вычислительной системы. Понятие ПО. Типы ПО. Понятие алгоритма. Свойства алгоритмов. Формы представления алгоритмов. Примеры записи алгоритмов. | 2 | - | - | 2 | - | 4 |
| 9 | **Архитектура ЭВМ.**  Понятие архитектуры ЭВМ. Классификация ЭВМ. Архитектура ЭВМ Фон Неймана. Принципы Фон Неймана. Шинная организация ЭВМ. Развитие шинной организация ЭВМ. Основной цикл работы ЭВМ. Организация ввода-вывода информации в ЭВМ. | 2 | 2 | - | 4 | 2 | 6 |
| 10 | **Процессоры и процессорные элементы вычислительных систем.**  Системы параллельной обработки данных. Процессоры с расширенной и сокращённой системой команд. Основные характеристики микропроцессоров, используемых в ПЭВМ.  Основные типы микропроцессоров, используемых в высокопроизводительных вычислительных системах. | 2 | - | - | 2 | - | 4 |
| 11 | **Хранение информации.**  Классификация запоминающих устройств. Основная память. Характеристики запоминающих устройств. Основные типы памяти современных ПК. Типы и структуры данных. Организация данных на устройствах с прямым и последовательным доступом. Контроль правильности работы запоминающих устройств. Внешние запоминающие устройства. Файлы данных. Файловые структуры. | 2 | - | - | 2 | - | 4 |
| 12 | **Информационный процесс накопления данных.**  Общая характеристика процессов накопления. Выбор хранимых данных. *Базы данных*. Реляционная модель БД. Реляционная структура данных. Целостность реляционных данных. | 2 | - | - | 2 | - | 4 |
| 13 | **Операционные системы**  ОС как интерфейс и как диспетчер. Режимы обработки данных: Пакетные режимы. Выполнение программы с прерываниями. Режим разделения времени. Режим реального времени. Планирование. Типы планирования. Алгоритмы планирования - однопроцессорные системы. Многопроцессорные системы и их варианты реализации. Алгоритм управления ресурсами - многопроцессорные системы (задача с прерываниями). Нетрадиционная обработка данных - параллельная обработка. Нетрадиционная обработка данных - последовательный конвейер. | 2 | - | - | 2 | - | 4 |
| 14 | **Информационный процесс обмена данными**  Общая схема системы передачи информации. Виды и характеристики носителей и сигналов. Спектры сигналов. Модуляция и кодирование. Каналы передачи данных и их характеристики. Методы повышения помехоустойчивости передачи и приема. Современные технические средства обмена данных и каналообразующей аппаратуры. Принципы помехоустойчивого кодирования. Циклические коды. Сжатие информации. | 2 | - | - | 2 | - | 4 |
| 15 | **Информационные сети.**  Классификация вычислительных сетей. Методы передачи данных по каналам связи. Способы коммутации данных. Модель взаимодействия открытых систем и протоколы обмена. Методы доступа к среде передачи данных. Сетевые технологии распределённой обработки данных. Структура Интернет. | 2 | - | - | 2 | - | 4 |
| 16 | **Контроль и защита информации в автоматизированных системах.**  Угрозы безопасности информации в АС. Обеспечение достоверности информации в АС. Обеспечение сохранности информации в АС. Обеспечение конфиденциальности информации в АС. Информационной безопасности от утечки по техническим каналам. Криптографическая защита. Система охраны объекта. Разграничение доступа в АС. | 2 | - | - | 2 | - | 4 |
| 17 | **Основы алгоритмизации.**  Способы представления алгоритмов. Графическое описание алгоритмов. Схема составления алгоритмов. Типы алгоритмов.  Поиск элемента в массиве: линейный, двоичный и интерполяционные алгоритмы. Поиск наибольшего и наименьшего элемента в массиве. Различные способы сортировки элементов массива. Разбиение и объединение строк, поиск и извлечение подстроки, удаление подстроки, синтаксический анализ текста.  Рекурсивные и итерационные алгоритмы. | - | 6 | 6 | 12 | 4 | 6 |
| 18 | **Программирование на языке С++ в среде Visual Studio**.  Интерфейс Visual Studio. Создание консольного проекта. Структура консольной программы на С++. Основные этапы написания программы. Программирование алгоритмов линейной структуры. Тестирование программы, точки останова и работа с отладчиком. Программирование алгоритмов разветвляющейся структуры. Логические условия. Циклы. Процедуры и функции. Использование встроенных библиотек.  Роль организации данных в программе. Представление текстовой информации. Различные способы организации строковых данных. Работа с отдельными символами. Наиболее употребительные функции для работы со строками. Применение списков и файлов.  Понятие массива. Массивы одномерные и многомерные. Понятие индекса и элемента массива. Типовые задачи с массивами. Пользовательский тип данных (структура). Объявление и использование пользовательского типа. | - | 20 | 22 | 44 | 5 | 16 |
|  | **Итого:** | **32** | **32** | **32** | **96** | **17** | **84** |
|  | **Экзамен 1 семестр** |  |  |  |  |  | **36** |
|  | **Экзамен 2 семестр** |  |  |  |  |  | **36** |

**7. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

1. Шишаев М.Г., Тоичкин Н.А. Архитектура (Организация) ЭВМ, (учебное пособие) Издательство КФПетрГУ, 2015.
2. Электронный образовательный ресурс «Информатика\_ИПМ» в системе MOODLE(модульная объектно-ориентированная динамическая учебная среда) МАГУ: <https://moodle.arcticsu.ru/course/view.php?id=170>.

**8. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

**Общие сведения**

|  |  |  |
| --- | --- | --- |
| 1. | Кафедра | **Информатики, вычислительной техники и информационной безопасности** |
| 2. | Направление подготовки | **09.03.02 «Информационные системы и технологии»** |
| 3. | Дисциплина (модуль) | **Б1.Б.5 Информатика** |

**Перечень компетенций**

|  |
| --- |
| * **ОПК-1** владение широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий; * **ОПК-5** способность использовать современные компьютерные технологии поиска информации для решения поставленной задачи, критического анализа этой информации и обоснования принятых идей и подходов к решению. |

**Критерии и показатели оценивания компетенций на различных этапах их формирования**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Этап формирования компетенции (разделы, темы дисциплины)** | **Формируемая компетенция** | **Критерии и показатели оценивания компетенций** | | | **Формы контроля сформированности компетенций** |
| **Знать:** | **Уметь:** | **Владеть:** |
| 1. *Введение в информатику.* | **ОПК-1**  **ОПК-5** | структуру и задачи информатики |  |  | Тест |
| 1. *Понятие информации и ее свойства.* | **ОПК-1**  **ОПК-5** | понятие информации и данных; основные свойства информации и способы ее измерения; тенденции роста информации в современном мире |  |  | Тест |
| 1. *Системы счисления* | **ОПК-1** | методику выполнять расчеты с числами представленными в различных системах счисления | выполнять решения задачи с двоичными и шестнадцатеричными числами; | методом перехода от десятичной системы исчисления к двоичной и шестнадцатеричной и наоборот | Тест, решение задач |
| 1. *Единицы представления информации в памяти ЭВМ.* | **ОПК-1** | принципы и методы представления информации различного вида в памяти ЭВМ; основные приемы используемые в алгоритмах сжатия данных | приводить двоичные числа к обратному и дополнительному коду | навыками счета в различных системах счисления | Тест, решение задач |
| 1. *Введение в алгебру логики.* | **ОПК-1** | основные законы алгебры логики | выполнять логический синтез переключательных вычислительных схем | навыками решения задач с логическими переменными и функциями | Тест, решение задач |
| 1. *Основы элементной базы цифровых автоматов.* | **ОПК-1** | основы схемотехники цифровых элементов |  |  | Тест, доклад |
| 1. *Элементы теории алгоритмов.* | **ОПК-1**  **ОПК-5** |  |  |  | доклад |
| 1. *Основы алгоритмизации.* | **ОПК-1**  **ОПК-5** | понятие и свойства алгоритма; основные алгоритмические структуры; основные современные средства разработки | выполнять алгоритмическую постановку задачи | навыками разработки блок-схем алгоритмов; навыками алгоритмизации; | Тест |
| 1. *Архитектура ЭВМ.* | **ОПК-1** | структуру и принципы организации работы современных ЭВМ | выбирать базовую конфигурацию компьютера | навыками работы на уровне продвинутого пользователя в операционной системе Windows | Тест |
| 1. *Процессоры и процессорные элементы вычислительных систем.* | **ОПК-1** | основные типы микропроцессоров, используемых в вычислительных системах |  |  | Тест, доклад |
| 1. *Хранение информации.* | **ОПК-1**  **ОПК-5** | классификация запоминающих устройств |  | навыками работы с запоминающими устройствами различных типов | Тест, доклад |
| 1. *Информационный процесс накопления данных.* | **ОПК-1**  **ОПК-5** | общую характеристика процессов накопления; понятие базы данных и различных моделей данных | организовывать реляционную структуру данных |  | Тест, доклад |
| 1. *Операционные системы* | **ОПК-1** | основные принципы функционирования операционных систем |  | навыками работы в операционной системе Windows | Тест, доклад |
| 1. *Информационный процесс обмена данными* | **ОПК-1** | общую схему системы передачи информации; современные технические средства обмена данных и каналообразующей аппаратуры |  |  | Тест, доклад |
| 1. *Информационные сети.* | **ОПК-1**  **ОПК-5** | понятие вычислительной сети; основные принципы организации локальных и глобальных вычислительных сетей |  |  | Тест, доклад |
| 1. *Контроль и защита информации в автоматизированных системах.* | **ОПК-1**  **ОПК-5** | основы информационной безопасности и сведений, составляющих государственную тайну; методы информационной безопасности; основы криптографии. |  |  | доклад |
| 1. *Основы алгоритмизации.* | **ОПК-1** | принципы разработки блок-схем алгоритмов | выполнять алгоритмическую постановку задачи | навыками разработки блок-схем алгоритмов; навыками алгоритмизации; | Тест, лабораторные работы |
| 1. *Программирование на языке С++ в среде Visual Studio.* | **ОПК-1**  **ОПК-5** | приемы разработки приложений в Visual Studio; основные алгоритмические конструкции языка С++; основные принципы разработки компьютерной графики | разрабатывать консольные приложения на языке программирования С++ | методикой создания консольных приложений с использованием среды Visual Studio;  методикой решения вычислительных задач на ЭВМ | Тест, лабораторные работы |

**Критерии и шкалы оценивания**

1. **Тест**

|  |  |  |  |
| --- | --- | --- | --- |
| Процент правильных ответов | До 60 | 61-80 | 81-100 |
| Количество баллов за решенный тест | 1 | 2 | 3 |

1. **Критерии оценки выступление студентов с докладом, рефератом, на семинарах**

|  |  |
| --- | --- |
| **Баллы** | **Характеристики ответа студента** |
| 5 | - студент глубоко и всесторонне усвоил проблему;  - уверенно, логично, последовательно и грамотно его излагает;  - опираясь на знания основной и дополнительной литературы, тесно привязывает усвоенные научные положения с практической деятельностью;  - умело обосновывает и аргументирует выдвигаемые им идеи;  - делает выводы и обобщения;  - свободно владеет понятиями |
| 3 | - студент твердо усвоил тему, грамотно и по существу излагает ее, опираясь на знания основной литературы;  - не допускает существенных неточностей;  - увязывает усвоенные знания с практической деятельностью;  - аргументирует научные положения;  - делает выводы и обобщения;  - владеет системой основных понятий |
| 2 | - тема раскрыта недостаточно четко и полно, то есть студент освоил проблему, по существу излагает ее, опираясь на знания только основной литературы;  - допускает несущественные ошибки и неточности;  - испытывает затруднения в практическом применении знаний;  - слабо аргументирует научные положения;  - затрудняется в формулировании выводов и обобщений;  - частично владеет системой понятий |
| **0** | - студент не усвоил значительной части проблемы;  - допускает существенные ошибки и неточности при рассмотрении ее;  - испытывает трудности в практическом применении знаний;  - не может аргументировать научные положения;  - не формулирует выводов и обобщений;  - не владеет понятийным аппаратом |

1. **Решение задач**

0,5 балла выставляется, если студент решил все рекомендованные задачи, правильно изложил все варианты их решения, аргументировав их, с обязательной ссылкой на соответствующие нормативы (если по содержанию это необходимо).

0,3 балла выставляется, если студент решил не менее 85% рекомендованных задач, правильно изложил все варианты решения, аргументировав их, с обязательной ссылкой на соответствующие нормативы (если по содержанию это необходимо).

0,2 балла выставляется, если студент решил не менее 65% рекомендованных задач, правильно изложил все варианты их решения, аргументировав их, с обязательной ссылкой на соответствующие нормативы (если по содержанию это необходимо).

0 баллов - если студент выполнил менее 50% задания, и/или неверно указал варианты решения.

1. **Выполнение лабораторной работы**

5 баллов выставляется, студент выполнил полностью все задания указанные в лабораторной работе и может аргументировано пояснить ход своего решения.

3 балла выставляется, если студент выполнил не менее 85 % заданий указанных в лабораторной работе, и может аргументировано пояснить ход своего решения и указать.

1 балл выставляется, если студент решил не менее 50% заданий указанных в лабораторной работе, и может аргументировано пояснить ход своего решения.

0 баллов выставляется, если студент не может аргументированно пояснить ход своего решения.

В случае если сроки сдачи работ превышены, количество баллов сокращается на 50%.

1. **Решение кейс-стади**

|  |  |
| --- | --- |
| **Баллы** | **Критерии оценивания** |
| 3 | * изложение материала логично, грамотно, без ошибок; * свободное владение профессиональной терминологией; * умение высказывать и обосновать свои суждения; * студент дает четкий, полный, правильный ответ на теоретические вопросы; * студент организует связь теории с практикой. |
| 2 | * студент грамотно излагает материал; ориентируется в материале, владеет профессиональной терминологией, осознанно применяет теоретические знания для решения кейса, но содержание и форма ответа имеют отдельные неточности; * ответ правильный, полный, с незначительными неточностями или недостаточно полный. |
| 1 | * студент излагает материал неполно, непоследовательно, допускает неточности в определении понятий, в применении знаний для решения кейса, не может доказательно обосновать свои суждения; * обнаруживается недостаточно глубокое понимание изученного материала. |
| 0 | * отсутствуют необходимые теоретические знания; допущены ошибки в определении понятий, искажен их смысл, не решен кейс; * в ответе студента проявляется незнание основного материала учебной программы, допускаются грубые ошибки в изложении, не может применять знания для решения кейса. |

***Типовые контрольные задания и методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы***

***1) Типовое тестовое задание***

1. Какой подход к измерению информации подразумевает использование понятия энтропии, как меры неопределенности состояния системы?

*1. Семантический (подход Шнайдера)*

*2. Прагматический*

*3. Статистический (подход Шеннона)*

2. При каком подходе к измерению информации используется тезаурусная мера?

*1. Семантический (подход Шнайдера)*

*2. Прагматический*

*3. Статистический (подход Шеннона)*

3. Какая элементная база использовалась в ЭВМ 1-го поколения?

1. *Транзисторы*
2. *Интегральные схемы*
3. *Электронные вакуумные лампы*

4. MIPS является единицей измерения

1. *объема памяти*
2. *производительности ЭВМ*
3. *частоты процессора*
4. К базовым алгоритмическим структурам относятся? 1) следование; 2) переход; 3) ветвление; 4) цикл; 5) передача
5. *1, 3, 4*
6. *2, 5*
7. *1, 2*
8. *3, 4, 5*

6. К основным структурам алгоритмов относятся: 1) линейные; 2) разветвляющиеся; 3) циклические; 4) графические; 5) повторяющиеся

1. *2, 5*
2. *1, 2, 3*
3. *1, 3, 4*
4. *4, 5*
5. Организация цикла, когда его тело расположено перед проверкой условия, носит название цикла с…
6. *Постусловием*
7. *Предусловием*
8. *Возвратом*
9. *Параметрами*
10. Цикл с \_\_\_\_\_\_\_\_\_ - цикл, при котором сначала вычисляется некоторое логическое выражение *Р*, в случае истинности которого выполняется тело цикла (оператор *S*).
11. *Предусловием*
12. *Постусловием*
13. *Переходом*
14. *Параметрами*

**Ключ:** 1-3, 2-1, 3-3, 4-2, 5-1, 6-2, 7-1, 8-1

***2) Примерные темы докладов***

1. История развития компьютерной техники.
2. Организация конвейера в вычислительных системах.
3. Суперконвейерные процессоры.
4. Суперскалярные процессоры.
5. Параллелизм в вычислительных системах.
6. Топологии вычислительных систем.
7. Потоковые вычислительные системы.
8. Редукционные вычислительные системы.
9. Векторные и векторно-конвейерные вычислительные системы.
10. Ассоциативные вычислительные системы.
11. Симметричные мультипроцессорные системы.
12. Кластерные вычислительные системы.
13. Вычислительны е системы на базе транспьютеров.
14. Управление памятью в операционных системах.
15. Особенности архитектуры микропроцессоров Intel 80x86 для организации мультипрограммных операционных систем.
16. Организация параллельных взаимодействующих процессов. Семафоры.
17. Проблема тупиков в операционных системах и методы борьбы с ними.
18. Операционные системы для мейнфреймов.
19. Операционные системы реального времени.
20. Операционные системы семейства UNIX.
21. Операционные системы семейства Windows.
22. История развития языков программирования.
23. Инструментальные среды программирования.
24. Инструментальные среды проектирования.
25. Математические программные пакеты.
26. Объектно-ориентированный подход в программировании.
27. Программирование в MS Office на MS Visual Basic.
28. Логическое и функциональное программирование.
29. Программирование в машинных кодах и ассемблеры.
30. Операционные системы для мобильных устройств.
31. Программирование для мобильных устройств.
32. Web – программирование и проектирование сайтов.
33. Векторная компьютерная графика.
34. Растровая компьютерная графика.
35. Визуализация данных научных исследований.
36. Моделирование данных и системы управления базами данных.
37. Разработка информационных систем основанных на знаниях.
38. Развивающие компьютерные игры.
39. Распределенная обработка данных.
40. Клиент-серверная архитектура вычислительных систем.
41. Корпоративные информационные системы и Intranet.
42. Технологии GRID.
43. Нано технологии в компьютерной технике.
44. Принципы построения компьютерных сетей.
45. Методы передачи дискретных данных на физическом уровне
46. Методы коммутации при передаче данных в компьютерных сетях.
47. Сетевой уровень модели OSI и маршрутизация пакетов.
48. Глобальные компьютерные сети.
49. Технологии мобильной связи.
50. Системы виртуальной реальности.
51. Многомерный анализ данных и OLAP – технология.
52. Онтологии и онтологические системы.
53. Искусственный интеллект и робототехника.
54. История развития и перспективы отечественной компьютерной техники.

***3) Пример решения задач***

**Пример 1.** Записать число -185 в двоичном виде.

*Решение:*

Для перевода отрицательных чисел в двоичный вил используется дополнительный код. Дополнительный код некоторого отрицательного числа представляет собой результат инвертирования (замены 1 на 0 и наоборот) каждого бита двоичного числа, равного модулю исходного отрицательного числа плюс единица.

Рассмотрим десятичное число –18510. Модуль данного числа в двоичном представлении равен 101110012.

**Первое** нужно дополнить это значение слева нулями до нужной размерности – байта, слова и т.д. Получим: 0000 0000 1011 10012.

**Второе** действие – получить *двоичное дополнение,* для этого все разряды двоичного числа необходимо инвертировать:

0000 0000 1011 10012 1111 1111 0100 01102.

**На третьем шаге** прибавляем единицу:

1111 1111 0100 01102 + 0000 0000 0000 00012 = 1111 1111 0100 01112

Результат этого преобразования равен 1111 1111 0100 01112 именно так и представляется число –18510 в памяти компьютера.

**Пример 2.** Используя метод Хаффмена построить двоичный код для приведенной таблицы частот:

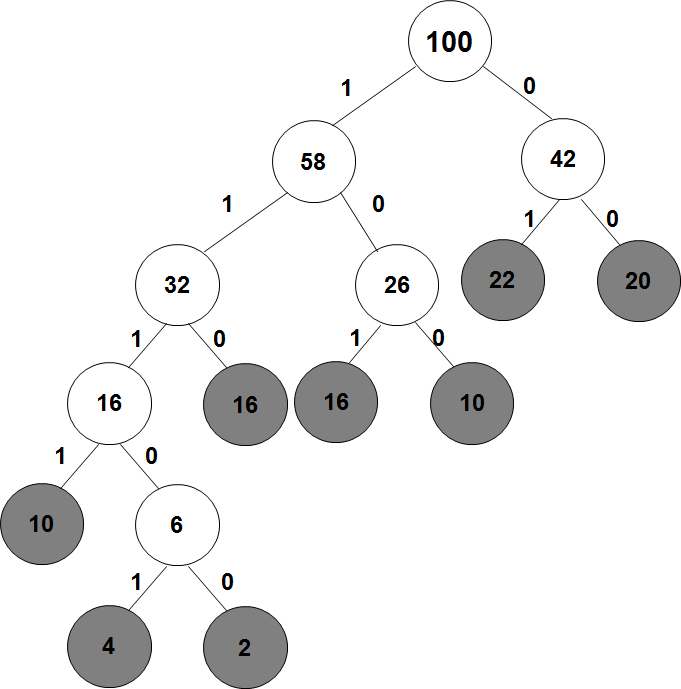
|  |  |
| --- | --- |
| ***Символы*** | ***Частоты*** |
| C | 22 |
| E | 20 |
| H | 16 |
| L | 16 |
| A | 10 |
| K | 10 |
| M | 4 |
| B | 2 |

*Решение*

Суть метода Хаффмена для построения двоичного кода сводится к следующему: символы встречающиеся в файле, выписывают в столбец в порядке убывания частоты их появления. Два последних символа объединяют в один с суммарной вероятностью. Из полученной новой вероятности и вероятностей новых символов, не использованных в объединении, формируется новый столбец в порядке убывания вероятностей, а две последние вновь объединяются. Это продолжается до тех пор пока не останется одна вероятность, равная сумме всех вероятностей всех символов, встречающихся в файле.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Символы*** | ***Частоты*** | ***Коды*** | | | | | | |
| C | 22 | 22 | 22 | 26 | 32 | 42 | 58 | 100 |
| E | 20 | 20 | 20 | 22 | 26 | 32 | 42 |  |
| H | 16 | 16 | 16 | 20 | 22 | 26 |  |  |
| L | 16 | 16 | 16 | 16 | 20 |  |  |  |
| A | 10 | 10 | 16 | 16 |  |  |  |  |
| K | 10 | 10 | 10 |  |  |  |  |  |
| M | 4 | 6 |  |  |  |  |  |  |
| B | 2 |  |  |  |  |  |  |  |

По данной таблице строится кодовое дерево: Из точки соответствующей сумме всех вероятностей, направляются две ветви. Ветви с большей вероятностью присваивается единица, с меньшей – вероятностью нуль. Далее последовательно продолжается процесс разветвления дерева, пока мы не доходим до вероятности каждого символа.



***Рис.1. Кодовое дерево Хаффмена для Примера 1***

После того как дерево построено, двигаясь по кодовому дереву сверху вниз, мы можем записать для каждого символа соответствующий код.

|  |  |  |
| --- | --- | --- |
| ***Символы*** | ***Вероятности*** | ***Коды*** |
| C | 22 | ***01*** |
| E | 20 | ***00*** |
| H | 16 | ***111*** |
| L | 16 | ***110*** |
| A | 10 | ***100*** |
| K | 10 | ***1011*** |
| M | 4 | ***10101*** |
| B | 2 | ***10100*** |

Для того чтобы обеспечить однозначное декодирование необходимо строить код так чтобы ни одна комбинация кода не совпадала с началом более длиной комбинации. Если код удовлетворяет этому условию, то он называется *префиксным*. Коды, построенные методом Шеннона – Фано или Хаффмена, являются префиксными.

***4) Пример задания на лабораторную работу***

**Тема: «Работа с массивами в языке программирования С++»**

# *Теоретические сведения*

Массив - это пронумерованная последовательность величин одинакового типа, обозначаемая одним именем. Элементы массива располагаются в последовательных ячейках памяти, обозначаются именем массива и индексом. Каждое из значений, составляющих массив, называется его *компонентой* (или *элементом* массива).

Массив данных в программе рассматривается как переменная структурированного типа. Массиву присваивается имя, посредством которого можно ссылаться как на массив данных в целом, так и на любую из его компонент.

Переменные, представляющие компоненты массивов, называются переменными с индексами в отличие от простых переменных, представляющих в программе элементарные данные. Индекс в обозначении компонент массивов может быть константой, переменной или выражением порядкового типа.

Если за каждым элементом массива закреплен только один его порядковый номер, то такой массив называется *линейным*. Вообще количество индексов элементов массива определяет *размерность* массива. По этом признаку массивы делятся на одномерные (линейные), двумерные, трёхмерные и т.д.

*Пример:* числовая последовательность четных натуральных чисел 2, 4, 6, ..., *N* представляет собой линейный массив, элементы которого можно обозначить А[1]=2, А[2]=4, А[3]=6, ..., А[К]=2\*(К+1), где *К* — номер элемента, а 2, 4, 6, ..., *N* — значения. Индекс (порядковый номер элемента) записывается в квадратных скобках после имени массива.

Например, A[7] — седьмой элемент массива А; D[6] — шестой элемент массива D.

Для размещения массива в памяти ЭВМ отводится поле памяти, размер которого определяется типом, длиной и количеством компонент массива.

тип идентификатор [количество строк];

Например, int B[5]; char R[34];

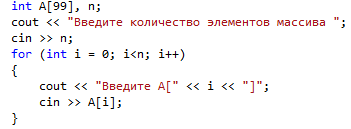
описывается массив *В*, состоящий из 5 элементов и символьный массив *R*, состоящий из 34 элементов. Для массива *В* будет выделено 5\*6=30 байт памяти, для массива *R*— 1\*34=34 байта памяти.

Тип элементов массива может быть любым, за исключением файлового. Обратим внимание, что нумерация элементов массива всегда идет с нуля.

Заполнить массив можно следующим образом:

Первый способ с помощью оператора присваивания. Этот способ заполнения элементов массива особенно удобен, когда между элементами существует какая-либо зависимость, например, арифметическая или геометрическая прогрессии, или элементы связаны между собой реккурентным соотношением.

Второй способ ввод значений элементов массива с клавиатуры используется обычно тогда, когда между элементами не наблюдается никакой зависимости. Например, последовательность чисел 1, 2, -5, 6, -111, 0 может быть введена в память следующим образом:



Над элементами массива чаще всего выполняются такие действия, как

а) поиск значений;

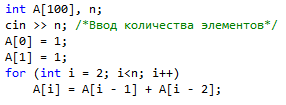
б) сортировка элементов в порядке возрастания или убывания;

в) подсчет элементов в массиве, удовлетворяющих заданному условию.

Cумму элементов массива можно подсчитать по формуле *S=S+A[I]* первоначально задав *S*=0. Количество элементов массива можно подсчитать по формуле *К*=*К*+1, первоначально задав *К*=0. Произведение элементов массива можно подсчитать по формуле   
*P = P \* A[I]*, первоначально задав *P* = 1.

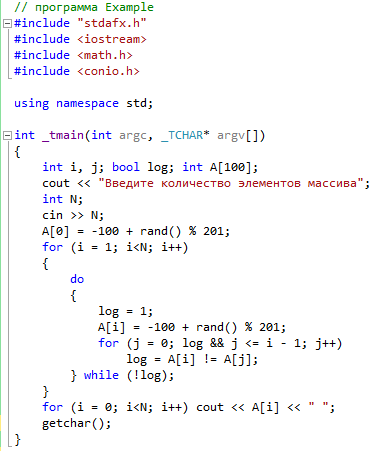
# *Примеры работы с массивами*

Задача 1**.** *Заполнить одномерный массив элементами, отвечающими следующему соотношению:* *a*1 = 1; *a*2 = 1; *a*i = *ai*-2 + a*i*-1 (*i* = 3, 4, ..., *n*).



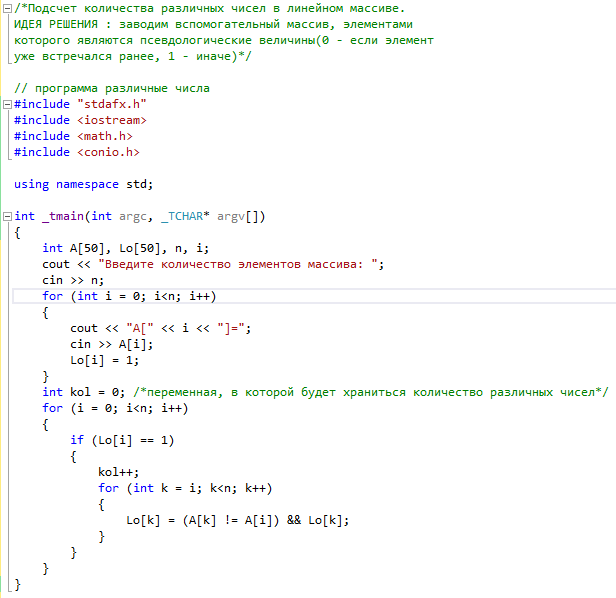
Другой вариант присваивания значений элементам массива — заполнение значениями, полученными с помощью датчика случайных чисел.

Задача 2. *Заполнить одномерный массив с помощью датчика случайных чисел таким образом, чтобы все его элементы были различны.*



|  |
| --- |
|  |

**Задача 3.** *Дан линейный массив целых чисел. Подсчитать, сколько в нем различных чисел.*



|  |
| --- |
|  |

# *Контрольные вопросы*

1. Что такое массив?
2. Почему массив является структурированным типом данных?
3. Что такое размерность массива? Существуют ли ограничения на размерность массива?
4. Какого типа могут быть элементы массива?
5. Какого типа могут быть индексы элементов массива?
6. Какими способами может быть заполнен массив? Приведите примеры.
7. Как определить минимальный объём памяти, отводимой под массив?
8. Какие действия выполняют обычно над элементами массива?
9. Может ли массив быть элементом массива?
10. Пусть элементами массива A (a[1], a[2], a[3], a[4]) являются соответственно *x*, *-x*, *x*2, -*x*2. Чему будет равно значение выражения: a[-a[a[3]-2]]+a[-a[a[3]]] при *x*=2?

***5). Пример кейс - стади***

Пусть необходимо выполнить покупку компьютера для одной из следующих целей:

* Рабочий компьютер для малого офиса.
* Компьютер для домохозяйки.
* Игровой компьютер.
* Компьютер-сервер для малого офиса.
* Компьютер-сервер для банка.
* Домашний компьютер для студента.

Необходимо определить конфигурацию аппаратного и программного обеспечения для представленных компьютеров.

Вопросы:

1. С чего необходимо начать выбор аппаратной части?
2. Какие требования к аппаратуре и программному обеспечению вы можете сформулировать для каждого из типов компьютера?
3. Оцените примерную стоимость выбранной вами конфигурации (используя данные из сети Интернет).
4. Каким образом необходимо проектировать конфигурацию вашего компьютера, для того чтобы сохранилась возможность его модификации?
5. Какое развитие конфигурации вы можете предложить?

**Вопросы к экзамену:**

**«Информатика» 1-ый семестр**

1. Понятие информации. Функции информации
2. Информационные процессы и системы. Основные этапы обращения информации в системах
3. Информационные процессы и системы. Классификации информационных систем. Понятие кибернетической системы.
4. Информационные ресурсы и технологии. Плюсы и минусы компьютеризации и информатизации общества.
5. Этапы развития задач хранения, обработки и передачи информации. Информатика как наука.
6. Структура науки информатика и ее связь с другими науками
7. Уровни проблем передачи информации. Меры информации на каждом уровне
8. Качество информации: совокупность свойств.
9. Виды и формы представления информации в информационных системах. Непрерывная и дискретная формы.
10. Системы счисления (СС). Виды СС и история их использования. Построение систем кодов на базе СС.
11. Позиционная система счисления (СС). Арифметические операции над числами в различных СС. Перевод чисел из одной СС в другую (на примере СС с основаниями 2, 8, 10, 16)
12. Представление числовой информации в цифровых автоматах (ЦА): ячейки памяти и регистры, электрические элементы и сигналы. Формы представления двоичных чисел в ЭВМ
13. Представление символьной информации в ЭВМ. Примеры систем кодировок и особенности их построения.
14. Представление графической информации в ЭВМ. Категории методов представления графической информации: представители этих категорий и их сравнение.
15. Обработка информации. Преобразование аналоговой информации в цифровую. Дискретизация, квантование непрерывного сигнала.
16. Алгебра логики, - как основа построения цифровых автоматов (дискретных вычислительных устройств). Основные элементы, операции и постулаты алгебры логики.
17. Представление функций алгебры логики. Операция суперпозиции функций.
18. Использование алгебры логики для разработки (синтеза) и анализа электрических переключательных схем вычислительных устройств. Функция проводимости. Этапы процедур синтеза и анализа.
19. Основные логические элементы современных вычислительных устройств: назначение, описание, условные обозначения. Триггеры.
20. Понятие алгоритма и алгоритмической системы. Свойства "интуитивного" понятия алгоритма. Язык алгоритма.
21. Математическое определение алгоритма через понятие "алфавитный оператор". Взаимосвязь и свойства алфавитных операторов и алгоритмов.
22. Общие (универсальные) способы задания алгоритмов. "Алгебраические" средства задания алгоритмов: машина Тьюринга.
23. Общие (универсальные) способы задания алгоритмов. "Геометрические" средства задания алгоритмов: блок-схемный метод алгоритмизации.
24. Компьютерная обработка информации. Формализация и абстракция. Исполнитель алгоритма. Связь ЭВМ и машины Тьюринга.
25. Основные операции при обработке информации на ЭВМ. Режимы организации вычислительного процесса в ЭВМ. Режимы взаимодействия пользователя с ЭВМ.
26. Организация взаимодействия пользователя и ЭВМ. Основные этапы реализации задач с помощью компьютера.

**Вопросы к экзамену:**

**«Информатика» 2-ой семестр**

1. Обработка информации. Классификация программного обеспечения. Системное ПО.
2. Обработка информации. Классификация программного обеспечения. Сервисное ПО.
3. Обработка информации. Классификация программного обеспечения. Инструментарий технологии программирования.
4. Обработка информации. Классификация программного обеспечения. Пакеты прикладных программ.
5. Обработка информации. Классификация средств обработки информации. Классификация ПК.
6. Обработка информации. Общая структура процессорных устройств. Принципы фон Неймана. Понятия архитектуры и структуры ПК. Общая схема ПК по фон Нейману.
7. Обработка информации. Общая структура процессорных устройств. Исполнение команд процессором. Форматы команд.
8. Общая структура ЭВМ. Основные понятия (персональный компьютер, процессор, память и т.д.). Блок-схема ПЭВМ.
9. Общая структура ЭВМ. Устройства ввода/вывода информации. Классификация устройств ввода информации.
10. Общая структура ЭВМ. Устройства ввода/вывода информации. Классификация устройств вывода информации.
11. Общая структура ЭВМ. Понятие магистрали/шины. Классификация шин. Примеры шин.
12. Системы параллельной обработки информации. Основные понятия. Способы реализации параллелизма. Векторная и конвейерная обработка информации.
13. Системы параллельной обработки информации. Классификация параллельных архитектур вычислительных систем (SIMD, MIMD, MISD, SISD).
14. Общая структура ЭВМ. Понятие системы команд процессора. Процессоры с расширенной и сокращенной системой команд.
15. Общая структура ЭВМ. Основные характеристики современных микропроцессоров. Понятия поколения и семейства процессора. Примеры процессоров.
16. Программная модель процессора. Регистры, классификация и их назначение.
17. Организация памяти, ее виды. Сегментная организация памяти.
18. Обработка прерываний. Определение прерывания, классификация прерываний. Система прерываний ПЭВМ.
19. Сетевые технологии. Архитектура «Клиент-Сервер», основные понятия и принципы. Классификация моделей распределенных вычислений.
20. Сетевые технологии. Всемирная паутина WWW, основные понятия и принципы. Способы реализации мигрирующих программ. Основные принципы технологий COM/DCOM, CORBA.
21. Хранение информации. Основные определения. Классификация памяти. Классификация запоминающих устройств. Типы современной основной памяти (SDRAM, DDRAM, DIMM и т.д.). Классификация внешних запоминающих устройств.
22. Передача информации. Основные термины. Общая схема передачи информации.
23. Передача информации. Виды и модели сигналов. Математическая модель сигналов и помех. Классификация помех.
24. Передача информации. Каналы передачи данных и их основные характеристики.
25. Передача информации. Классификация линий связи. Кабельные линии связи.
26. Передача информации. Классификация линий связи Беспроводные линии связи.
27. Передача информации. Аппаратура линий связи. Классификация аппаратуры линий связи.
28. Информационные сети. Понятия телекоммуникационной, информационной и вычислительной сети. Классификации по разным критериям вычислительных сетей.
29. Информационные сети. Методы передачи данных по каналам связи, режимы передачи.
30. Информационные сети. Способы коммутации данных (коммутация каналов, коммутация пакетов, коммутация сообщений).
31. Информационные сети. Эталонная модель взаимодействия открытых систем: структура уровней и их назначение. Основные протоколы.
32. Контроль и защита информации в автоматизированных системах. Основные понятия и термины. Классификация угроз безопасности.

**9. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)**

*а*) основная литература:

# [Моисеев Н. Г.](https://biblioclub.ru/index.php?page=author_red&id=156860) Теория автоматов: учебное пособие по курсовому проектированию, Йошкар-Ола: [ПГТУ](https://biblioclub.ru/index.php?page=publisher_red&pub_id=17562), 2015, 127 с.

# [Грузина Э. Э.](https://biblioclub.ru/index.php?page=author_red&id=80746) , [Корчуганова М. Р.](https://biblioclub.ru/index.php?page=author_red&id=80747) Компьютерные науки: учебное пособие, Ч. I, Кемерово: [Кемеровский государственный университет](https://biblioclub.ru/index.php?page=publisher_red&pub_id=15522), 2009, 130 с.

*б) дополнительная литература:*

# [Прохорова О. В.](https://biblioclub.ru/index.php?page=author_red&id=96440) Информатика: учебник, Самара: [Самарский государственный архитектурно-строительный университет](https://biblioclub.ru/index.php?page=publisher_red&pub_id=5121), 2013, 106 с.

# Бородихин В.Н. Языки программирования (Си/Си++): учебно-методическое пособие, Омск: [Омский государственный университет](https://biblioclub.ru/index.php?page=publisher_red&pub_id=1067), 2013, 200 с.

# [Мейер Б.](https://biblioclub.ru/index.php?page=author_red&id=152044) Инструменты, алгоритмы и структуры данных. М.: [Национальный Открытый Университет «ИНТУИТ»](https://biblioclub.ru/index.php?page=publisher_red&pub_id=16364), 2016, 543 с.

**10. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ "ИНТЕРНЕТ" (ДАЛЕЕ - СЕТЬ "ИНТЕРНЕТ"), НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)**

* 1. <http://www.intuit.ru> – национальный открытый университет.
  2. <http://citforum.ru> – справочная информация по различным разделам информационных технологий.

**11. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

Приступая к изучению дисциплины, студенту необходимо внимательно ознакомиться с тематическим планом занятий, списком рекомендованной литературы. Следует уяснить последовательность выполнения индивидуальных учебных заданий. Самостоятельная работа студента предполагает работу с научной и учебной литературой. Уровень и глубина усвоения дисциплины зависят от активной и систематической работы на лекциях, изучения рекомендованной литературы, выполнения контрольных письменных заданий.

Студентам рекомендуется регулярно посещать лекции, тщательно конспектировать и прорабатывать их с одним из рекомендованных литературных источников.

При изучении курса студентам рекомендуется проводить:

* конспектирование первоисточников и другой учебной литературы;
* проработку учебного материала (по конспектам, учебной и научной литературе);
* подготовку к коллоквиумам с использованием электронных методических указаний по курсу;
* самотестирование по контрольным вопросам (тестам).

Каждый студент перед началом занятий записывается преподавателем на электронный курс по данному предмету, к которому можно получить доступ через сеть Интернет. Курс поддерживается системой дистанционного обучения *moodle* (модульная объектно-ориентированная динамическая учебная среда), к которой может получить доступ зарегистрированный пользователь через сеть Интернет. Адрес курса в системе *moodle* МАГУ: <http://moodle.arcticsu.ru/course/view.php?id=170> [[1]](#footnote-1).

В рамках данного курса в системе *moodle*, организовано:

* взаимодействие обучающихся между собой и с преподавателем: для чего используются форумы и чаты.
* передача знаний в электронном виде: с помощью файлов, архивов, веб-страниц, лекций.
* проверка знаний и обучение с помощью тестов и заданий:  результаты работы студенты могут отправлять в текстовом виде или в виде файлов.
* совместная учебная и исследовательская работа студентов по определенной теме: с помощью встроенных механизмов: семинаров, форумов и пр.
* журнал оценок: в котором учитывается успеваемость студентов по балльной системе.

Таким образом, вся самостоятельная работа студента организуется через систему дистанционного обучения *moodle* МАГУ. Так же данная система используется преподавателем и в процессе проведения аудиторных занятий, для: проведения тестов, предоставления текстов лекций и лабораторных работ, учета успеваемости учащихся.

Основными видами аудиторной работы студентов являются лекции и практические работы.

В ходе лекций преподаватель излагает и разъясняет основные, наиболее сложные понятия темы, а также связанные с ней теоретические и практические проблемы, дает рекомендации на лабораторные работы и указания на самостоятельную работу. Электронные конспекты презентаций и полных текстов лекций доступны для студента в системе *moodle* МАГУ на сайте курса: «Информатика (физическое направление)».

На практических и лабораторных занятиях студенты совместно с преподавателем обсуждают выданные им проектные задания, задают интересующие их вопросы и выполняют на компьютерах самостоятельно или в группах свои проекты, используя программное обеспечение представленной в рабочей программе. Каждое выполненное задание студент обязан оформить в виде отчета и защитить его. Электронные конспекты лабораторных заданий доступны для студента в системе *moodle* МАГУ на сайте курса: «Информатика\_ИПМ».

Качество учебной работы студентов преподаватель оценивает с использованием технологической карты дисциплины, размещенной на сайте МАГУ.

**Планы практических занятий:**

**1-ый семестр**

**Занятие 1. Представление информации в цифровом автомате**

**План:**

1. Системы счисления.
2. Перевод чисел из одной позиционной системы счисления в другую.
3. Формы представления чисел.
4. Определение погрешностей.

*Литература*: [1, с. 10-29].

*Вопросы для самоконтроля*

1. Дайте определение системы счисления.
2. В чем состоит отличие позиционной системы счисления от непозиционной?
3. Что такое базис в позиционной системе счисления?
4. Запишите эквиваленты десятичных цифр в системах счисления с основанием 2,8,16.
5. Запишите формулу перевода числа из одной позиционной системы счисления в другую?
6. Что такое естественная и нормальная форма записи числа?
7. Что означает – представление числа с фиксированной запятой?
8. Что означает – представление числа с плавающей запятой?
9. Что такое мантисса числа?
10. Как оценить погрешности возникающие при переводе чисел из одной системы счисления в другую?

*Задание для самостоятельной работы*

* 1. Представить заданное десятичное число в различных системах счисления.
  2. Перевести делением заданное целое десятичное число в двоичную систему счисления.
  3. Перевести число, заданное в 2-ой системе счисления, в десятичную.
  4. Перевести число, заданное в 8-ой системе счисления, в десятичную.
  5. Перевести число, заданное в 16-ой системе счисления, в десятичную.
  6. Перевести десятичную дробь в двоичную дробь.
  7. Подготовиться к ответам на вопросы.

**Занятие 2. Сложение, умножение и деление чисел на двоичных сумматорах**

**План:**

1. Формальные правила двоичной арифметики.
2. Двоичный полусумматор.
3. Двоичный сумматор.
4. Представление отрицательных чисел.
5. Сложение чисел в форме с фиксированной запятой на двоичном сумматоре.
6. Переполнение разрядной сетки.
7. Особенности сложения чисел в форме с плавающей запятой.
8. Основные методы выполнения операции умножения в двоичной системе счисления.
9. Умножение чисел в форме с фиксированной запятой на различных типах двоичного сумматора.
10. Особенности умножения чисел в форме с плавающей запятой.
11. Методы выполнения операции деления.
12. Деление чисел, в форме с фиксированной запятой с восстановлением остатков и без.
13. Особенности деления чисел в форме с плавающей запятой.

*Литература*: [1, с. 30-55].

*Вопросы для самоконтроля*

1. Запишите правила выполнения арифметических действий двоичного полусумматора.
2. Запишите правила выполнения арифметических действий двоичного сумматора.
3. Представьте условные обозначение двоичных сумматоров и полусумматоров.
4. Запишите правила вычитания двоичных цифр в двоичном вычитателе.
5. Что такое прямой код числа?
6. Что такое дополнительный код числа?
7. Приведите алгоритм получения дополнительного кода для отрицательного числа.
8. Что такое обратный код числа?
9. Дайте определение двоичного сумматора прямого кода.
10. Дайте определение двоичного сумматора дополнительного кода.
11. В чем состоит особенность двоичного сумматора дополнительного кода?
12. Дайте определение двоичного сумматора обратного кода.
13. Что является признаком переполнения разрядной сетки для различных сумматоров?
14. Что такое разряд переполнения?
15. В чем состоят особенности сложения чисел в форме с плавающей запятой?
16. В чем состоит операция нормализации числа?

*Задание для самостоятельной работы*

* 1. Представить заданное двоичное число в прямом коде.
  2. Представить заданное двоичное число в дополнительном коде.
  3. Представить заданное двоичное число в обратном коде.
  4. Сложить два числа на двоичном сумматоре прямого кода.
  5. Сложить два числа на двоичном сумматоре дополнительного кода.
  6. Сложить два числа на двоичном сумматоре обратного кода.
  7. Сложить два числа в форме с плавающей запятой на сумматорах дополнительного кода.
  8. Сложить два числа в форме с плавающей запятой на сумматорах обратного кода.
  9. Подготовиться к ответам на вопросы.

**Занятие 3. Умножение чисел на двоичных сумматорах**

**План:**

1. Основные методы выполнения операции умножения в двоичной системе счисления.
2. Умножение чисел в форме с фиксированной запятой на различных типах двоичного сумматора.
3. Особенности умножения чисел в форме с плавающей запятой.
4. Способы ускорения операций умножения.

*Литература*: [1, с. 55-70].

*Вопросы для самоконтроля*

* + 1. Перечислите основные способы выполнения операции умножения в двоичной системе счисления.
    2. Приведите структурную схему множительного устройства по каждой из схем умножения в двоичной системе.
    3. В чем состоят особенности перемножения двоичных чисел в форме с плавающей запятой.
    4. Какие особые случаи могут возникнуть при выполнении операции умножения.
    5. Приведите методы ускорения операции умножения? В чем они заключаются?

*Задание для самостоятельной работы*

1. Найти произведение двоичных чисел, в форме с фиксированной запятой на сумматоре прямого кода.
2. Найти произведение двоичных чисел, в форме с фиксированной запятой на сумматоре дополнительного кода.
3. Найти произведение двоичных чисел, в форме с фиксированной запятой на сумматоре обратного кода.
4. Найти произведение двоичных чисел, в форме с плавающей запятой.
5. Подготовиться к ответам на вопросы.

**Занятие 4. Деление чисел на двоичных сумматорах**

**План:**

1. Методы выполнения операции деления.
2. Деление чисел, в форме с фиксированной запятой с восстановлением остатков.
3. Деление чисел, в форме с фиксированной запятой без восстановления остатков.
4. Особенности деления чисел в форме с плавающей запятой.
5. Способы ускорения операций деления.

*Литература*: [1, с. 71-81].

*Вопросы для самоконтроля*

1. Какие существуют методы выполнения операции деления?
2. В чем заключается алгоритм деления с восстановлением остатка?
3. На сумматорах, какого типа может быть реализован алгоритм деления с восстановлением остатка?
4. В чем заключается алгоритм деления с восстановлением остатка?
   * 1. В чем состоят особенности деления двоичных чисел в форме с плавающей запятой.
     2. Приведите методы ускорения операции деления? В чем они заключаются?

*Задание для самостоятельной работы*

1. Выполнить деление чисел, в форме с фиксированной запятой, с восстановлением остатка.
2. Выполнить деление чисел, в форме с фиксированной запятой, без восстановления остатка.
3. Выполнить деление двоичных чисел, в форме с плавающей запятой.
4. Подготовиться к ответам на вопросы.

**2-ой семестр**

**Занятие 1. Логические основы построения ЭВМ**

**План:**

1. Базовые логические функции.
2. Логические выражения.
3. Таблицы истинности.
4. Основные законы алгебры логики (булевой алгебры).
5. Минимизация логических выражений.
6. Карты Карно.
7. Безразличные значения

*Литература*: [2, с. 26-30].

*Вопросы для самоконтроля*

1. Что такое таблица истинности.
2. Постройте таблицу истинности для логической функции И.
3. Постройте таблицу истинности для логической функции ИЛИ.
4. Для чего нужны карты Карно?

*Задание для самостоятельной работы*

* 1. Докажите заданное логическое тождество.
  2. Постройте таблицу истинности для заданной логической функции.
  3. Выполните минимизацию логического выражения.
  4. Выполните минимизацию логического выражения, используя карты Карно.
  5. Подготовиться к ответам на вопросы.

**Занятие 2. Разработка логических электронных схем в MMLogic**

**План:**

1. Базовые логические вентили.
2. Синтез базовых логических вентилей.
3. Вентили: И-НЕ, ИЛИ-НЕ. Синтез вентилей И-НЕ, ИЛИ-НЕ.
4. Построение схем триггера, сумматора, полусумматора.
5. Принципы работы в среде MMlogic

*Литература*: [2, с. 31-34].

*Вопросы для самоконтроля*

1. Перечислите базовые логические вентили и их графическое изображение.
2. Покажите, обладают ли вентили И-НЕ свойством ассоциативности?

*Задание для самостоятельной работы*

* + 1. Постройте заданную логическую функцию, используя базовые вентили. Протестируйте ее работу.
    2. Постройте функцию «Исключающее ИЛИ» используя только двухвходовые вентили И-НЕ.
    3. Перевести десятичную дробь в двоичную дробь.
    4. Подготовиться к ответам на вопросы.

**Занятие 3. Рекурсия в программировании**

**План:**

1. Рекурсивное определение. Рекурсивные выражения.
2. Рекурсивно определенные структуры данных.
3. Рекурсивно определяемые алгоритмы и программы.
4. Рекурсивная и не рекурсивная версия чисел Фибоначчи.
5. Прямая и косвенная рекурсии.
6. Пример рекурсии: Ханойская башня.
7. Построение дерева рекурсивных вызовов.
8. Рекурсия как стратегия решения задач.
9. Сравнение рекурсии и итерации.

*Литература*: [5, с. 243-260].

*Вопросы для самоконтроля*

1. Что такое рекурсивное определение?
2. Приведите пример рекурсивного алгоритма.
3. В каком случае работа рекурсивного алгоритма уйдет в бесконечность?
4. В чем отличие прямой рекурсии от косвенной.
5. Приведите пример рекурсивного типа данных?
6. Какая реализация алгоритма лучше с точки зрения времени исполнения: рекурсивная или не рекурсивная?
7. Что такое глубина рекурсии?
8. Чему равна сложность рекурсивного алгоритма: «Ханойские башни»?

*Задание для самостоятельной работы*

* + 1. Напишите на одном из языков программирования рекурсивную и не рекурсивную реализацию алгоритма: «Числа Фибоначчи».
    2. Напишите на одном из языков программирования рекурсивную реализацию алгоритма: «Ханойские башни».
    3. Постройте дерево рекурсивных вызовов для рекурсивной реализации алгоритма: «Ханойские башни» при n = 3.
    4. Постройте дерево рекурсивных вызовов для рекурсивной реализации алгоритма: «Ханойские башни» при n = 4.
    5. Постройте дерево рекурсивных вызовов для заданного рекурсивного алгоритма.
    6. Напишите на одном из языков программирования процедуру вывода чисел от 1 до 100 без использования цикла.
    7. Подготовиться к ответам на вопросы.

**Планы лабораторных работ:**

**1-ый семестр**

**Лабораторная работа № 1.** **Условные операторы и операторы цикла в С++**

**План:**

1. Работа с величинами.
2. Ввод/вывод.
3. Общая структура программы на С++.
4. Операции языка С++.
5. Преобразования типов.
6. Математические и логические функции.
7. Примеры программ.

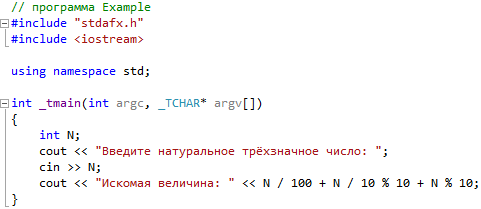
*Литература*: [4, с. 7-17].

*Вопросы для самоконтроля*

1. Что такое величина?
2. Какие величины называют аргументами? результатами? промежуточными величинами? Приведите примеры.
3. Каковы атрибуты величины?
4. Какие величины называют постоянными? переменными? Приведите примеры.
5. Какие простые типы величин существуют в языке C++?
6. Что определяет тип величины?
7. Расскажите о простых типах данных и их атрибутах.
8. Как осуществляется потоковый ввод данных в языке C++? Приведите примеры.
9. Как осуществляется потоковый вывод данных в языке C++? Приведите примеры.
10. Какова общая структура программы в языке C++?
11. Расскажите об операторе присваивания и преобразовании типов.
12. Что такое формат вывода?
13. Расскажите о правилах вычисления алгебраического выражения. Приведите примеры.
14. Расскажите о правилах вычисления логического выражения. Приведите примеры.
15. Расскажите о логических операциях. Приведите примеры.
16. Приведите примеры задач, имеющих линейный алгоритм решения.
17. Выпишите несколько алгебраических выражений и запишите их на языке C++.

*Задание для самостоятельной работы*

1. Каково назначение следующей программы?



1. Задайте на координатной плоскости некоторую область, которую можно описать математическими уравнениями и заштрихуйте её. Запишите логическое выражение, принимающее значение 1, если точка (*x*, *y*) лежит внутри заштрихованной области, иначе - 0.
2. Написать программу, на языке С++, вычисляющую формулу математического выражения, приведенного ниже в вариантах задания. Ввод данных организовать с клавиатуры, вывод данных на экран.
3. Подготовиться к ответам на вопросы.

**Лабораторная работа № 2. Введение в программирование на языке C++. Линейные алгоритмы.**

**План:**

1. Условный оператор. Полная и неполная развилка. Примеры.
2. Переключатель. Примеры.
3. Циклы с параметром. Примеры.
4. Итерационные циклы (с пред- или постусловием). Примеры.
5. Сложные циклы. Примеры.

*Литература*: [4, с. 18-26].

*Вопросы для самоконтроля*

1. Когда возникает необходимость в организации развилки?
2. Какая развилка называется полной? неполной?
3. Выражение какого типа может выступать в качестве условия при организации развилки? Какие значения принимают такие выражения?
4. Могут ли в полной развилке не выполниться операторы ни по одной из ветвей? выполниться по обеим ветвям?
5. Записать примеры 1-3 по теме "переключатель" с помощью условного оператора. Сколько развилок понадобилось в каждом из случаев?
6. В каком случае целесообразно использовать переключатель?
7. Используя переключатель, решить задачу: "Определить знак заданного целого числа".
8. Назовите отличия итерационных циклов и цикла с параметром.
9. Какова структура оператора цикла с параметром? Как выполняется цикл с параметром?
10. Могут ли параметр цикла, его начальное и конечное значения в цикле с параметром в языке C++ быть разных типов? Обоснуйте ответ.
11. Может ли один цикл быть вложен внутрь другого? Если да, то какова глубина этой вложенности?
12. Какова структура циклов с пред- и постусловием? как выполняются эти циклы?
13. Каково минимальное и максимальное количество исполнений циклов с пред- и постусловием? С чем это связано?
14. Для цикла с параметром запишите его полный эквивалент с помощью циклов с пред- и постусловием.
15. Для цикла с предусловием запишите его полный эквивалент с помощью цикла с постусловием.
16. Для цикла с постусловием запишите его полный эквивалент с помощью цикла с предусловием.

*Задание для самостоятельной работы*

1. Сколько раз исполнится фрагмент программы?

for (i = 1; i <= -1; i++) k = k \* i;

1. Сколько раз исполнится фрагмент программы?

for (i = -1; i <= 1; i++) k = k \* i;

1. Сколько раз исполнится фрагмент программы?

for (i = 1; i <= -1; i--) k = k \* i;

1. Сколько раз исполнится фрагмент программы?

M = 123; while (M != 0) M = M % 10;

1. Написать программу, на языке С++, табулирования функции, указанной в варианте задания, на заданном отрезке с заданным шагом.
2. Подготовиться к ответам на вопросы.

**Лабораторная работа № 3.** **Одномерные массивы в С++**

**План:**

1. Задание одномерных массивов. Примеры.
2. Заполнение массивов с помощью датчика случайных чисел.
3. Примеры обработки массивов.

*Литература*: [4, с. 27-35].

*Вопросы для самоконтроля*

1. Что такое массив?
2. Почему массив является структурированным типом данных?
3. Что такое размерность массива? Существуют ли ограничения на размерность массива?
4. Какого типа могут быть элементы массива?
5. Какого типа могут быть индексы элементов массива?
6. Какими способами может быть заполнен массив? Приведите примеры.
7. Как определить минимальный объём памяти, отводимой под массив?
8. Какие действия выполняют обычно над элементами массива?
9. Может ли массив быть элементом массива?

*Задание для самостоятельной работы*

* 1. Пусть элементами массива A (a[1], a[2], a[3], a[4]) являются соответственно *x*, *-x*, *x*2, -*x*2. Чему будет равно значение выражения: a[-a[a[3]-2]]+a[-a[a[3]]] при *x*=2?
  2. Написать программу, на языке С++, обработки элементов массива согласно варианту задания. Элементы массива вводятся с помощью датчика случайных чисел.
  3. Подготовиться к ответам на вопросы.

**2-ой семестр**

**Лабораторная работа № 1.** **Двумерные массивы в С++**

**План:**

1. Задание двумерных массивов. Примеры.
2. Примеры обработки массивов.

*Литература*: [4, с. 27-35].

*Вопросы для самоконтроля*

1. Можно ли выполнять обход двумерного массива, организовав внешний цикл по столбцам, а внутренний — по строкам?
2. Используются ли вложенные циклы, если совершается обход только главной диагонали квадратной матрицы?

*Задание для самостоятельной работы*

* + 1. Написать программу, на языке С++, обработки элементов двумерного массива согласно варианту задания. Элементы массива вводятся с помощью датчика случайных чисел.
    2. Подготовиться к ответам на вопросы.

**Лабораторная работа № 2. Подпрограммы в С++**

**План:**

1. Подпрограммы: процедуры и функции. Примеры.
2. Объявление и описание функций.
3. Фактические и формальные параметры функций.
4. Вызовы функций.
5. Прототипы функций.
6. Возврат значений. Ключевое слово return.
7. Функция main.
8. Примеры разработки функций в С++.

*Литература*: [4, с. 43-51].

*Вопросы для самоконтроля*

1. Какие алгоритмы называют вспомогательными?
2. Какое количество вспомогательных алгоритмов может присутствовать в основном алгоритме?
3. Можно ли вспомогательные алгоритмы, написанные для решения данной задачи, использовать при решении других задач, где их применение было бы целесообразно?
4. Какие параметры называют формальными? фактическими?
5. Какое соответствие должно соблюдаться между формальными и фактическими параметрами?
6. Может ли фактических параметров процедуры (функции) быть больше, чем формальных? А меньше?
7. Существуют ли подпрограммы без параметров?
8. Существуют ли ограничения на число параметров подпрограмм? Если нет, то чем же всё-таки ограничивается это количество в С++?
9. В каком разделе объявляются и в каком реализуются подпрограммы в С++?
10. Какого типа может быть значение функции?
11. Расскажите о методе последовательной детализации при разработке программ.

*Задание для самостоятельной работы*

1. Написать программу, на языке С++, в которой реализованы функции заданные согласно варианту задания.
2. Подготовиться к ответам на вопросы.

**Лабораторная работа № 3. Работа с одномерными массивами с использованием процедур**

**План:**

1. Передача одномерного массива в функцию.
2. Задание динамического массива (одномерного).
3. Передача динамического массива в функцию.

*Литература*: [4, с. 52-57].

*Вопросы для самоконтроля*

1. Какие массивы называют динамическими?
2. Как задается одномерный динамически массив.
3. Каким образом происходит выделение памяти под динамический массив?
4. Как в С++ динамический массив передается в функцию?

*Задание для самостоятельной работы*

1. Написать программу, на языке С++, согласно варианту задания: каждый пункт задания оформить в виде функции. Все необходимые данные для функции должны передаваться ей в качестве параметров. Использовать только динамические массивы.
2. Подготовиться к ответам на вопросы.

**Лабораторная работа № 4. Строки в С++. Работа с файлами в С++**

**План:**

1. Строки в С++.
2. Функции для работы со строками и символами.
3. Работа с файлами в С++.
4. Режимы открытия файлов.

*Литература*: [4, с. 36-42]; [4, с. 74-94].

*Вопросы для самоконтроля*

1. Каким образом представляются с троки в С++?
2. Приведите примеры объявление строки.
3. Обязательно ли необходимо инициализировать строку, при ее объявлении?
4. Какая функция организует побайтовое копирование символов из одной строки в другую?
5. Приведите пример функции для объединения строк.
6. Приведите пример функции для сравнения строк.
7. Какой заголовочный файл необходимо подключить для работы с файлами?
8. Какие действия необходимо выполнить для того чтобы создать текстовый файл и записать в него строку?
9. Какие действия необходимо выполнить для того чтобы прочитать строку из текстового файла?
10. Перечислите режимы открытия файлов.

*Задание для самостоятельной работы*

1. Написать программу, на языке С++, согласно варианту задания. Программа генерирует текстовый файл, читает его, выполняет определенные действия и  выполняет вывод на консоль и в файл.
2. Подготовиться к ответам на вопросы.

**12. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ), ВКЛЮЧАЯ ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ (ПРИ НЕОБХОДИМОСТИ)**

*Программное обеспечение:*

1. Microsoft Windows XP – операционная система.
2. Open Office Writer – текстовый редактор.
3. Open Office Calc – электронная таблица.
4. MultiMedia Logic.
5. Microsoft Visual Studio – среда разработки программных приложений.
6. Любой web-браузер.

**Тренажеры:** В целях обучения студентов, усвоения и контроля полученных знаний используется электронный образовательный ресурс moodle: (<http://moodle.arcticsu.ru/>) в котором студенты могут проходить тесты. Адрес курса: <http://moodle.arcticsu.ru/course/view.php?id=170>

**13. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

|  |  |  |
| --- | --- | --- |
| № п\п | Наименование оборудованных учебных кабинетов, объектов для проведения занятий с перечнем основного оборудования | Фактический адрес учебных кабинетов и объектов, номер ауд. |
| 1 | Кабинет для проведения лекционных занятий по курсу с комплектом мультимедийного оборудования, включающий ноутбук, мультимедиапроектор и экран | г. Апатиты, ул. ул. Лесная, 29 корпус 5, ауд. 106, 302 |
| 2 | Компьютерный класс для проведения лабораторных занятий и тестирования. | г. Апатиты, ул. Энергетическая, 19, корпус 5, ауд. 4 |

**14. ТЕХНОЛОГИЧЕСКАЯ КАРТА ДИСЦИПЛИНЫ**

**ОСНОВНАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА**

**09.03.02 «Информационные системы и технологии» профиль: Общий**

(код, направление, профиль)

**ТЕХНОЛОГИЧЕСКАЯ КАРТА**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Шифр дисциплины по РУП | | | | | | | **Б1.Б.5** | | | | | | | | | | |
| Дисциплина | | | | **Информатика** | | | | | | | | | | | | | |
| Курс | **1** | | семестр | | | **1** | | | | | | | | | | | |
| Кафедра | | **Информатики, вычислительной техники и информационной безопасности** | | | | | | | | | | | | | | | |
| Ф.И.О. преподавателя, звание, должность | | | | | | | | | | **Тоичкин Николай Александрович, к.т.н., доцент** | | | | | | | |
|  | | | | | | | | | | | | | | | | | |
| Общ. трудоемкостьчас/ЗЕТ | | | | | | **144/4** | | Кол-во семестров | | | | **1** | Интерактивные формыобщ./тек. сем. | | | | **17/0** |
| ЛКобщ./тек. сем. | | | **16/16** | | ПР/СМобщ./тек. сем. | | | | **16/16** | | ЛБобщ./тек. сем. | | | **16/16** | Форма контроля | **Экзамен** | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Содержание задания** | **Количество мероприятий** | **Максимальное количество  баллов** | **Срок предоставления** |
| ***Вводный блок*** | | | |
| Не предусмотрен |  |  |  |
| ***Основной блок*** | | | |
| Решение бланочных тестов | 5 | 15 | 2 неделя  5 неделя  8 неделя  11 неделя  15 неделя |
| Лабораторные и практические работы | 7 | 35 | 2 неделя  4 неделя  6 неделя  8 неделя  10 неделя  12 неделя  15 неделя |
| Подготовка докладов по теме | 1 | 5 | По согласованию с преподавателем |
| Решение комплекса задач | 2 | 5 | 4 неделя |
| **Всего:** | | **60** |  |
| Экзамен | Вопрос 1 | 20 | В сроки сессии |
|  | Вопрос 2 | 20 | В сроки сессии |
| **Всего:** | | **40** |  |
| **Итого:** | | **100** |  |
| ***Дополнительный блок*** | | | |
| Выполнение дополнительной лабораторной работы | | 10 | по согласованию с преподавателем |
| Подготовка глоссария | | 5 |
| **Всего баллов по дополнительному блоку:** | | **15** |

Шкала оценивая в рамках балльно-рейтинговой системы МАГУ: «2» - 60 баллов и менее, «3» - 61-80 баллов, «4» - 81-90 баллов, «5» - 91-100 баллов.

**ОСНОВНАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА**

**09.03.02 «Информационные системы и технологии» профиль: Общий**

(код, направление, профиль)

**ТЕХНОЛОГИЧЕСКАЯ КАРТА**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Шифр дисциплины по РУП | | | | | | | **Б1.Б.5** | | | | | | | | | | |
| Дисциплина | | | | **Информатика** | | | | | | | | | | | | | |
| Курс | **1** | | семестр | | | **2** | | | | | | | | | | | |
| Кафедра | | **Информатики, вычислительной техники и информационной безопасности** | | | | | | | | | | | | | | | |
| Ф.И.О. преподавателя, звание, должность | | | | | | | | | | **Тоичкин Николай Александрович, к.т.н., доцент** | | | | | | | |
|  | | | | | | | | | | | | | | | | | |
| Общ. трудоемкостьчас/ЗЕТ | | | | | | **144/4** | | Кол-во семестров | | | | **1** | Интерактивные формыобщ./тек. сем. | | | | **17/17** |
| ЛКобщ./тек. сем. | | | **16/16** | | ПР/СМобщ./тек. сем. | | | | **16/16** | | ЛБобщ./тек. сем. | | | **16/16** | Форма контроля | **Экзамен** | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Содержание задания** | **Количество мероприятий** | **Максимальное количество  баллов** | **Срок предоставления** |
| ***Вводный блок*** | | | |
| Не предусмотрен |  |  |  |
| ***Основной блок*** | | | |
| Решение бланочных тестов | 5 | 15 | 2 неделя  5 неделя  8 неделя  11 неделя  15 неделя |
| Лабораторные и практические работы | 7 | 35 | 2 неделя  4 неделя  6 неделя  8 неделя  10 неделя  12 неделя  15 неделя |
| Подготовка докладов по теме | 1 | 5 | По согласованию с преподавателем |
| Решение кейсов | 2 | 5 | 14 неделя |
| **Всего:** | | **60** |  |
| Экзамен | Вопрос 1 | 20 | В сроки сессии |
|  | Вопрос 2 | 20 | В сроки сессии |
| **Всего:** | | **40** |  |
| **Итого:** | | **100** |  |
| ***Дополнительный блок*** | | | |
| Выполнение дополнительной лабораторной работы | | 10 | по согласованию с преподавателем |
| Подготовка глоссария | | 5 |
| **Всего баллов по дополнительному блоку:** | | **15** |

Шкала оценивая в рамках балльно-рейтинговой системы МАГУ: «2» - 60 баллов и менее, «3» - 61-80 баллов, «4» - 81-90 баллов, «5» - 91-100 баллов.

**15. ИНЫЕ СВЕДЕНИЯ И МАТЕРИАЛЫ НА УСМОТРЕНИЕ ВЕДУЩЕЙ КАФЕДРЫ**

Не предусмотрено.

**16. ОБЕСПЕЧЕНИЕ ОБРАЗОВАНИЯ ДЛЯ ЛИЦ С ОВЗ**

Для обеспечения образования инвалидов и лиц с ограниченными возможностями здоровья реализация дисциплины *Б1.Б.5 «Информатика»* может осуществляться в адаптированном виде, с учетом специфики освоения и дидактических требований, исходя из индивидуальных возможностей и по личному заявлению обучающегося.

1. Ссылка активна на момент написания рабочей программы (30.09.16). Для получения доступа к курсу необходима регистрация в системе и запись на курс. [↑](#footnote-ref-1)