

Введение

[1. Цели и задачи курса «Основы алгоритмизации и программирования»](#)

[2. Mind- карта курса](#)

[3. Обзор курса «Основы алгоритмизации и программирования»](#)

[4. О системе рейтингового контроля по дисциплине](#)

1. Цели и задачи курса «Основы алгоритмизации и программирования»

Целью курса «Основы алгоритмизации и программирования» является освоение студентами навыков алгоритмизации и программирования задач для решения их на современных вычислительных машинах.

Неотъемлемой чертой современного специалиста в области компьютерных технологий является умение разрабатывать алгоритмы и реализовывать их на компьютере, алгоритмически подходить к решению информационных задач, разбираться в терминологии программирования, представлять возможности современных языков разработки программного обеспечения.

В настоящее время существует большое количество разнообразных языков программирования, с помощью которых можно эффективно решать широкий круг задач. Но залогом успешной разработки программного обеспечения на любом языке программирования было и остается знание основных принципов алгоритмизации, понимание процесса работы программы, обработки компьютером машинных команд. Это является базисом для программиста любого профиля.

Отдельное внимание на занятиях будет уделено различным способам организации данных в программе, решению стандартных алгоритмических задач. Рассматриваемые вопросы являются основополагающими и для дисциплин «Конструирование программ и языки программирования», «Объектно-ориентированное программирование» и других, которые студенты будут изучать позже.

Материал курса использует знания, полученные студентам при изучении высшей математики: математические ряды, методы нахождения корней уравнений, системы уравнений, численные методы, приближенное вычисление функций.

В результате освоения курса «Основы алгоритмизации и программирования» студент должен:

знать:

- теоретические основы алгоритмизации задач и проектирования программ;
- основы организации вычислительных процессов на ЭВМ;
- приемы проектирования на современном алгоритмическом языке высокого уровня;

уметь характеризовать:

- исходные данные решаемых задач;
- выходные данные решаемых задач и формы их представления;

уметь анализировать:

- решаемые задачи и четко формулировать их условия;
- возможные методы решения задач и осуществлять их обоснованный выбор;

приобрести навыки:

- формальной (математической) постановки задач;
- алгоритмизации задач;
- программирования, отладки и выполнения на ЭВМ конкретных задач с использованием современных методов программирования.

2. Mind- карта курса

Mind- картой всего учебного курса или его части мы будем называть схематическое изображение содержания в виде звездообразной структуры. Как утверждают психологи, такая схема помогает уяснить существующие логические связи между разделами курса и облегчает понимание и запоминание материала курса. Ниже приводится сокращенная mind-карта лекционной части курса «Основы алгоритмизации и программирования» с указанием числа часов, отводимых на различные разделы курса.



3. Обзор курса «Основы алгоритмизации и программирования»

Курс «Основы алгоритмизации и программирования» построен по модульному принципу и включает двенадцать модулей.

Первый модуль- «Введение»- содержит описание целей и задач курса, mind- карту курса, а также обзор всех модулей. Кроме того, во введении описывается рейтинговая система контроля знаний слушателей, позволяющая оценить успешность освоения данного курса.

Второй модуль курса посвящен рассмотрению общих сведений об алгоритмах и цифровых вычислительных машинах. Дано определение понятию «алгоритм», рассмотрены основные свойства алгоритмов. Также рассматривается структурная схема типовой цифровой вычислительной машины с точки зрения программиста, использующего ЦВМ для реализации разработанных им алгоритмов.

В третьем модуле дано описание языка программирования Pascal. Рассмотрены основные конструкции языка, стандартные процедуры ввода- вывода, структура программы, стандартные функции, реализованные в языке. Приведена классификация типов переменных, определенных в языке.

Четвертый модуль содержит описание характерных приемов алгоритмизации. Рассмотрены типовые задачи- табуляция функций, приближенное вычисление определенных интегралов, рекуррентные вычисления, цепные дроби.

В пятом модуле рассматриваются процедуры- универсальное средство повышения уровня программирования. Описаны такие важные понятия как локальность, формальные и фактические параметры процедур. Рассмотрены способы организации процедур- функций.

Значительное внимание в данном модуле уделяется такому важному понятию программирования как рекурсия. В завершение описывается технология пошаговой детализации программ с использованием процедур.

Шестой модуль посвящен описанию структурных типов данных. Здесь же рассматриваются массивы- важнейшие представители этого типа. Описываются типовые алгоритмы обработки массивов, а также многомерные массивы.

Седьмой модуль описывает такие типы данных, как строки, записи и множества, а также операторы, предназначенные для обработки данных этих типов.

В восьмом блоке вводится понятие программного модуля. Дается описание организации модулей в системе Turbo Pascal. Также описываются стандартные модули системы программирования Turbo Pascal- Crt и Graph.

Девятый блок посвящен рассмотрению приемов работы со структурой данных «Таблица». Описываются реализованные на таблице методы поиска нужной информации, а также методы сортировки таблиц.

Десятый блок содержит описание файлов, а также стандартных процедур и функций для обработки файлов, реализованных в языке программирования Pascal.

Одиннадцатый блок посвящен описанию способов хранения информации в оперативной памяти компьютера. Рассматриваются такие понятия как статическая память, автоматическая память, динамическая память. Описываются способы создания динамических структур данных в среде программирования Pascal.

В двенадцатом блоке рассматриваются основы объектно- ориентированного программирования.

4. О системе рейтингового контроля по дисциплине

Рейтинговая система оценки знаний позволяет снять недостатки традиционной системы контроля или, по крайней мере, нейтрализовать некоторые из них. Она предназначена для организации учебно-воспитательного процесса на базе новейших технологий обучения, активизирующих самостоятельное творческое мышление учащихся и стимулирующих рост профессионального мастерства педагогического состава.

Применение данной системы в учебном процессе позволяет:

- получить объективную оценку знаний учащегося в процессе обучения, так как отражает динамику усвоения знаний по предмету за семестр и является суммарным коэффициентом, а не количественной и качественной характеристикой усвоенных знаний непосредственно на момент проверки знаний, как в традиционном обучении.
- повысить эффективность самостоятельной работы учащегося, обеспечить ее ритмичность в течении семестра, в отличие от традиционного обучения, где учащийся может быть пассивным в течении семестра и «просыпается» только при подготовке к экзамену или зачету;
- ранжировать учащихся по уровню знаний и определить их дальнейший путь обучения;
- повысить эффективность работы преподавателей;
- более эффективно использовать стимуляцию для активизации работы преподавателей и учащихся.

Для курса «Основы алгоритмизации программирования» принята 100-балльная шкала рейтинга. Это значит, что при условии успешного прохождения всех этапов рубежного контроля и стопроцентной посещаемости лекционных и лабораторных занятий студент может набрать к концу семестра 100 баллов.

Баллы, оценивающие различные виды деятельности учащегося приведены в следующей таблице:

100% посещение лекций и лабораторных занятий	Одна задача на экзамене и на аттестационной контрольной работе	Критерий 1-ой аттестации	Критерий 2-ой аттестации	Одна задача повышенной сложности на экзамене
5 баллов	5 баллов	20 баллов	40 баллов	40 баллов

Графиком учебного процесса в семестре предусмотрено две аттестации- на 7-ой и 14-ой неделях семестра соответственно. Аттестации проводятся в виде контрольных работ, на которых студенту предлагается для решения 5 задач. Так как каждая задача оценивается в 5 баллов (см. таблицу), в случае успешного решения всех пяти задач студент набирает 25 баллов. Если же к тому же у данного студента не было пропусков занятий, то к текущему рейтингу в 25 баллов добавляется еще и 5 баллов за 100% посещение занятий. Таким образом, после первой аттестации его текущий рейтинг будет равняться 30 баллам. Аналогично рассчитываются баллы и после второй аттестации. Это значит, что успешно занимающийся студент, посетивший 100% лекционных и лабораторных занятий, может подойти к экзамену с рейтингом 60 баллов.

Экзамен состоит из двух частей- теста, проверяющего знание теоретической части материала и практической части, проверяющей умение решать задачи. По результатам прохождения теста принимается решение о допуске студента к практической части. При этом тест считается пройденным, если студент правильно ответил не менее чем на 70% вопросов теста. Если тест пройден успешно, студенту предлагается пять задач обычной и две задачи повышенной сложности. Время, отводимое для решения задач ограничено, и составляет 2 астрономических часа. По истечении этого времени, в соответствии с вышеприведенной таблицей, подсчитываются набранные студентом баллы, которые суммируются с текущим

рейтингом и образуют итоговый рейтинг по дисциплине. Если, например, во время экзамена студентом решена задача повышенной сложности, оцениваемая в 40 баллов, а текущий рейтинг до экзамена составлял 60 баллов, то итоговый рейтинг составит 100 баллов, что эквивалентно оценке «десять». Вообще, переход от баллов рейтинга к экзаменационным оценкам осуществляется просто- рейтинг делится на 10 и результат округляется путём отбрасывания дробной части.