

УТВЕРЖДАЮ

Проректор по учебной работе  
Ю. А. Самарский

10 июня 2010 г.

**ПРОГРАММА**

по дисциплине **ОСНОВЫ ИНФОРМАТИКИ** (алгоритмы и алгоритмические языки). Основной курс  
по направлению подготовки: 010900 «Прикладные математика и физика» факультеты: **ФРТК, ФОПФ, ФАКИ, ФМБФ, ФФКЭ. ФАЛТ, ФУПМ**

кафедра: информатики

курс: I

семестр: I

Зачетные единицы - 5

Трудоемкость: базовая часть - 3 зач. ед.;

вариативная часть - 2 зач. ед.,

в т.ч. по выбору студента - 0 зач. ед.:

лекции: - 34 (час.)

практические (семинарские) Экзамен - нет

занятия: - 34 (час.) Зачет дифф. - 1 сем.

лабораторные занятия: - 34 (час.) Самостоятельная работа

**ВСЕГО АУДИТОРНЫХ ЧАСОВ – 102**

Программу составили: академик РАН В. П. Иванников,  
доцент, к.ф.-м.н. А. В. Ворожцов,  
доцент, к.ф.-м.н. П. Н. Коротин

Программа обсуждена на заседании кафедры информатики 21 мая 2010 г.

Заведующий кафедрой  
д.ф.-м.н., профессор

И. Б. Петров

Компетенции обучающегося, формируемые в результате освоения дисциплины:

ОК-10, ОК-11, ОК-12, ОК-13, ПК-6, ПК-12, ПК-14 Структура преподавания дисциплины

**Введение в теорию алгоритмов.** Интуитивное понятие алгоритма. Свойства алгоритмов. Понятие об исполнителе алгоритма. Алгоритм как преобразование слов из заданного алфавита. Связь понятия алгоритма с понятием функции. Машина Тьюринга. Нормальные алгорифмы Маркова. Вычислимые функции и их свойства. Невычислимые функции. Различные эквивалентные определения множества вычислимых функций. Алгоритмическая сложность.

**Алгоритмические языки.** Характеристика алгоритмических языков и их исполнителей. Понятие трансляции.

Понятие о формальных языках. Способы строгого описания формальных языков, понятие о метаязыках. Алфавит, синтаксис и семантика алгоритмического языка. Описание синтаксиса языка с помощью металингвистических формул и синтаксических диаграмм

Языки программирования. Общие характеристики языков программирования. Алфавит, имена, служебные слова, стандартные имена, числа, текстовые константы, разделители.

Типы данных, их классификация Переменные и константы. Скалярные типы данных и операции над ними. Старшинство операций, стандартные функции. Выражения и правила их вычисления. Оператор присваивания. Перечислимые и ограниченные типы данных.

Файлы. Стандартные функции ввода-вывода.

Простые и сложные операторы. Пустой, составной, условный операторы. Оператор варианта. Оператор перехода.

Оператор цикла. Программирование рекуррентных соотношений.

Составные типы данных. Массивы.

Описание функций (процедур) Формальные и фактические параметры. Способы передачи параметров. Локализация имен. Побочные эффекты. Итерации и рекурсии.

Ссылочный тип данных. Методы выделения памяти: статический, динамический и автоматический.

**Этапы разработки программ.** Документирование, тестирование и верификация программного кода. Различные примеры модуляризации программных систем (библиотеки, программные интерфейсы, методы взаимодействия модулей).

Учебно-методическое и информационное обеспечение дисциплины

Основная литература

1. *Ворожцов А. В., Винокуров Н. А.* Практика и теория программирования. - М.: Физматкнига, 2008.
2. *Керниган Б. У, Ритчи Д. М.* Язык программирования С. -2-е изд. - М.: Издательский дом «Вильямс», 2006.

Дополнительная литература

1. *Шилдт Г.* Полный справочник по С. - М.: Издательский дом «Вильямс», 2005.
2. *Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г.* Теория и реализация языков программирования. - М.: МЗ Пресс, 2006.
3. *Митницкий В. Я.* Элементы теории алгоритмов и язык программирования С. - М : МФТИ, 2001.

Пособия и методические указания

1. *Прут В. В.* Введение. Целые и рациональные числа: методические указания к практикуму по курсу Основы информатики. - М: МФТИ, 2002.
2. *Прут В. В.* Матрицы. Геометрия. Фракталы. Игры: методические указания к практикуму по курсу Основы информатики. - М.: МФТИ, 2002.
3. *Прут В. В.* Математическая логика. Теория алгоритмов. Рекурсия. Сортировка. Графы: методические указания к практикуму по курсу Основы информатики. - М.: МФТИ, 2002.

Электронные ресурсы

1. <http://cs.mipt.L4i>
2. <http://acm.mipt.ru>

**ЗАДАНИЕ 1**

(срок сдачи 26-31 октября)

1. Машины Тьюринга

Обозначим как  $N_0$  множество всех неотрицательных целых чисел. Описать машины Тьюринга, которые реализуют:

1.1. Счетчик четности. Выход машины Тьюринга равен 0 или 1 в зависимости от того, четно или нечетно число единиц в последовательности из 0 и 1, записанной на ленте машины Тьюринга. В конце последовательности стоит символ  $B$ . В начальном состоянии головка видит первый левый символ.

1.2. Инверсию заданного слова в алфавите  $\{0, 1\}$  (0 заменяет на 1, а 1 - на 0).

1.3. «Переворачивание» заданного слова в алфавите  $\{a, b, c\}$ .

1.4. Сложение двух чисел из множества  $N_0$ , записанных на ленте в виде последовательности единиц, а именно:

$0 \rightarrow 0, 1 \rightarrow 01, 2 \rightarrow 011, 3 \rightarrow 0111, 4 \rightarrow 01111, \dots$  Назовем эту запись *единичной записью* числа. Записи чисел разделены на ленте несколькими пустыми ячейками. Рассмотрите различные варианты начального положения головки машины Тьюринга.

1.5. Сложение двух чисел из  $N_0$ , данных в двоичной системе счисления.

1.6. Распознавание правильных скобочных выражений. Правильное скобочное выражение - это слово в алфавите  $A = \{(\,)\}$ , которое может получиться, если из арифметического выражения удалить все символы, кроме скобок. Примеры правильных скобочных выражений: пустое слово,  $()$ ,  $((\ ))$ ,  $00((\ ))$ . Примеры неправильных скобочных выражений:  $)$ ,  $(\ )$ ,  $((\ ))$ ,  $((\ ))$ ,  $((\ ))$ . Результат работы: слово «YES», если скобочно, выражение правильное, и слово «NO» - иначе.

1.7. Перемножение двух чисел из  $N_0$ , заданных в виде единичных записей. Числа записаны на ленте подряд. 1.8 Вычисление квадрата числа, заданного в виде единичной записи.

## 2. Алгоритмы Маркова

2.1. Записать нормальные алгоритмы Маркова, которые реализуют:

2.1.1 Приписывание буквы  $X$  к входному слову справа.

2.1.2. Задание 1.2.

2.1.3. Задание 1.3.

2.1.4. Задание 1.5.

2.1.5. Задание 1.6.

2.1.6. Удвоение числа, заданного а) в виде единичной записи, б) в двоичной системе счисления.

2.2. В алфавите  $A = \{a, b\}$  описать нормальный алгоритм, который выдает в качестве результата пустое слово, если буквы  $a$  и  $b$  входят во входное слово в равном количестве, и любое непустое слово - иначе. В алгоритме должно быть не более четырех правил подстановки. Докажите правильность придуманного алгоритма.

2.3. Существует ли алгоритм Маркова, применимый только к двоичным записям простых чисел?

2.4. Верно ли, что человек для любого алгоритма Маркова может написать машину Тьюринга, реализующую ту же функцию (то есть множество слов, к которым они применимы, совпадает и для любого элемента из этого множества результаты работы алгоритма Маркова и машины Тьюринга совпадают)? Разрешима ли эта задача алгоритмически?

2.5. Верно ли, что человек для любого алгоритма Маркова и заданного входного слова может определить, применим алгоритм Маркова к этому слову или нет? Можно ли поручить эту программу компьютеру (в принципе, не принимая во внимание доступное время и вычислительные мощности)?

## ЗАДАНИЕ 2

*(срок сдачи 14-19 декабря)*

### **3. Решение простых алгоритмических задач**

При написании программ в качестве входного и выходного потоков использовать стандартные потоки ввода и вывода. При решении задач аккуратно форматируйте код согласно правилам. Давайте переменным и функциям значимые имена. Прежде чем сдавать программу преподавателю, проверьте, что она правильно работает на более чем 10 различных входных данных.

В каждой задаче ответьте на вопрос о том, как растет время работы программы и используемая программой память с ростом параметра размера входных данных (например, параметра  $n$ ).

3.1. «Мах». Написать программу, которая выводит максимальное число из  $n$  заданных чисел. В первой строчке входа дано число  $i$ , а в следующей строчке указано  $n$  целых чисел.

3.2. «Числа Фибоначчи I». Написать программу, которая по данному  $n$  находит  $n$ -е число Фибоначчи  $F_n$ . Числа Фибоначчи определяются соотношениями:

$$F_n = F_{n-1} + F_{n-2}, \quad F_1 = F_2 = 1$$

Подсказка: организуйте цикл, в котором по последним двум вычисленным числам будет вычисляться следующее. Необходимо ли хранить в памяти все вычисленные числа Фибоначчи?

3.3. «Числа Фибоначчи II». Решить предыдущую задачу, используя идею рекурсии. Оценить число элементарных операций, которое необходимо сделать в рекурсивном и нерекурсивном алгоритме вычисления числа  $F_n$ .

3.4. «Биномиальные коэффициенты». Написать программу, которая для данного натурального числа  $n$  находит коэффициенты в разложении

$$(1+x)^n = C_n^0 x^0 + C_n^1 x^1 + \dots + C_n^n x^n.$$

Использовать соотношения

$$C_n^n = C_n^0 = 1, \quad C_n^k = C_{n-1}^k + C_{n-1}^{k-1}.$$

Оценить, как растет время работы вашей программы с ростом  $n$ .

3.5. «Простые числа». Написать программу, которая определяет, является ли введенное число  $n$  простым.

3.6. «Скобки». Написать программу, которая определяет, является ли введенная скобочная структура правильной (см. задание 1.6).

Ответьте на те же вопросы при условии, что максимальное число правил ограничено числом 10.

3.7. «Задача Иосифа». Пусть  $p$  человек, стоящие по кругу, считаются (начиная с первого) считалкой из  $m$  слов. Человек, на котором считалка заканчивается, - выбывает, круг смыкается, а счет продолжается с человека, следующего за выбывшим. Напишите программу, выводящую номера трех человек, выбывших последними, в порядке их выбывания. При написании программы следует использовать динамические переменные.

3.8. «Уравнение». Написать программу, которая находит корни уравнения  $e^x = x + 2$  с погрешностью  $10^{10}$ . Сколько итераций необходимо сделать, чтобы достичь указанной точности методом деления пополам?

3.9. «Лабиринт». Используя рекурсию, написать программу, которая находит кратчайший путь из левого верхнего угла лабиринта в нижний правый угол либо определяет, что такого пути нет. Лабиринт задан в виде прямоугольного клеточного поля, белые клетки в котором считаются проходимыми, а черные

- непроходимыми. В первой строчке входа заданы размеры лабиринта  $N$  и  $M$  по горизонтали и вертикали соответственно. Далее идут  $N$  строчек, в каждой из которых дано слово в алфавите  $\{\#, .\}$  из  $M$  символов. Символ '#' означает черную клетку, а '.' - белую. Выход программы должен содержать последовательность пар координат клеток, по которым нужно идти, либо слово «NO».

3.10. В строках текстового файла содержатся разделенные пробелом целочисленные координаты вершин прямоугольной односвязной плоской фигуры. Вершины фигуры перечисляются в произвольном порядке. Вычислить площадь заданной фигуры.

3.11. На поле СЗ шахматной доски неподвижно располагается белый король. Произвольно на доску выставляются черный король и белый ферзь. Напишите программу, доказывающую, что при любой (имеющей смысл) расстановке двух последних фигур мат черному королю может быть объявлен не более чем за 25 ходов.