

УТВЕРЖДЕНО
Проректор по учебной работе
и довузовской подготовке
А. А. Воронов
27 июня 2017 г.

ПРОГРАММА

по дисциплине: **ИНФОРМАТИКА. Основной курс**
по направлению подготовки: 03.03.01 «Прикладная математика и физика»
физтех-школа: **ФФПФ**
факультет: **ФОПФ**
кафедра: **информатики и вычислительной математики**
курс: 1
семестр: 1

Трудоемкость:

Базовая часть – 4 зачет. ед.:

лекции: – 30 часов

практические (семинарские)

занятия: – нет

лабораторные занятия: – 60 часов

Экзамен – нет

Зачет дифф. – 1 семестр

Две контрольные работы

Самостоятельная работа – 90 часов

ВСЕГО АУДИТОРНЫХ ЧАСОВ – 90

Программу составили: академик РАН В. П. Иванников,
к.ф.-м.н., доцент П. Н. Коротин,
д.ф.-м.н., профессор В. В. Прут

Программа принята на заседании
кафедры информатики и вычислительной математики
16 июня 2017 г.

Заведующий кафедрой,
чл.-корр. РАН

И. Б. Петров

Структура преподавания дисциплины.

1. Язык программирования С. История языка. Объекты языка. Константы. Переменные. Типы данных. Массивы. Указатели. Массивы указателей. Указатели и многомерные массивы. Наборы символов ASCII, Unicode; кодовые страницы. Символьные и строковые типы данных. Структуры. Битовые поля. Объединения. Инициализация всех типов переменных. Преобразование типов. Определение размера данных. Операции: арифметические, логические, отношения. Приоритеты операций. Операторы присваивания, условных и безусловных переходов, цикла, дополнительные. Структура программы. Функции. Способы передачи параметров. Прототипы функций. Указатели на функции. Рекурсия. Библиотечные функции. Функции ввода-вывода. Файлы. Классы памяти. Области действия переменных. Динамическое распределение памяти. Динамические матрицы. Оператор определения типа. Препроцессор. Включение файлов. Макроподстановка. Программирование в графическом режиме. Методы отладки программ. Рекомендуемая среда программирования: Microsoft Visual Studio.
2. Рекурсивные и итеративные алгоритмы. Индукция и рекурсия. Рекурсивные методы в программировании. Примеры рекурсивных подпрограмм.
3. Классические алгоритмы. Перебор и методы его сокращения, перебор с возвратом. Динамическое программирование. “Жадные” алгоритмы.
4. Алгоритмы арифметики. Системы счисления. Наибольший общий делитель. Простые числа: генерация и проверка на простоту. Каноническое разложение. Алгоритмы для больших чисел.
5. Алгоритмы сортировки данных. Сортировки: включением, выбором, обменом. Быстрая сортировка. Использование быстрой сортировки для разных типов данных. Пирамидальная сортировка. Сортировка слиянием. Поразрядная сортировка. Лексикографическая сортировка. Внешняя сортировка. Сравнение методов сортировки.
6. Структуры данных. Основные определения. Представление в памяти. Связанные списки. Стеки. Очереди. Очереди с приоритетом. Способы реализации. Функции обработки и примеры использования структур данных. Сортировка списков. Алгоритмы преобразования инфиксной формы записи в постфиксную и префиксную, обратная операция. Вычисление выражений.
7. Деревья. Способы представления в памяти. Базовые операции и их реализация. Способы обхода двоичного дерева, их рекурсивные и итеративные реализации. Дерево выражений, построение и обработка. Сортировка с помощью дерева.
8. Алгоритмы поиска. Последовательный и бинарный поиск. Бинарное дерево поиска (BST). Базовые операции и их реализация. Вставка узла в

лист и корень. Удаление узла. Объединение деревьев. Вращение дерева. Сбалансированные BST деревья. AVL-дерево. Хеширование. Методы построения хеш-функций. Разрешение коллизий. Сравнение скоростей поиска.

9. Алгоритмы на графах. Представление графов. Поиск в глубину и в ширину. Кратчайшие пути. Эйлеров и гамильтонов циклы. Раскраска вершин, ребер и граней графа.

10. Комбинаторные алгоритмы. Генерация комбинаторных объектов: перестановок, сочетаний, размещений. Генерация разбиений числа.

11. Алгоритмы алгебры логики. Элементарные логические операции. Таблицы истинности. Алгебраические свойства элементарных операций. Нормальные формы. Полные системы функций. Базисы.

12. Алгоритмы вычислений на строках. Точное совпадение: классические методы. Приближенный поиск подстрок. Регулярные выражения.

13. Алгоритмические основы машинной графики.

14. Основы программирования на Python. Вводный курс по программированию, дающий представление о базовых понятиях языка.

15. Компьютерное моделирование в физике.

16. Сложность вычислений. NP-полнота. Труднорешаемые задачи. Эвристические алгоритмы.

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 C. – 2-е изд. – М.: Издательский Дом «Вильямс», 2006.

3. *Прут В. В.* Алгоритмы и структуры данных на языке C. Учебное пособие. – М.: МФТИ, 2016.

Дополнительная литература

1. *Прага С.* Язык программирования C. – М.: Издательский Дом «Вильямс», 2006.

2. *Шилдт Г.* Полный справочник по C. – М.: Издательский Дом «Вильямс», 2005.

3. *Ахо А., Хопкрофт Дж., Ульман Дж.* Структуры данных и алгоритмы. М.: Издательский Дом «Вильямс», 2000.

4. *Вирт Н.* Алгоритмы и структуры данных. – СПб.: Невский Диалект, 2005.

5. *Новиков Ф. А.* Дискретная математика для программистов. СПб.: Питер. 2001.

6. *Кормен Т. Х., Лейзерсон Ч. И., Ривест Р. Л., Штайн К.* Алгоритмы: построение и анализ. 3-е изд. – М.: Издательский Дом «Вильямс», 2013.

7. *Седжвик Р.* Алгоритмы на C++. – М.: Издательский Дом «Вильямс», 2011.
8. *Шевелев Ю. П.* Дискретная математика. – СПб.: изд-во «Лань», 2008.
9. *Гашков С. Б., Чубариков В. Н.* Арифметика. Алгоритмы. Сложность вычислений. М.: Изд. МГУ, «Дрофа», 2005.
10. *Смит Б.* Методы и алгоритмы вычислений на строках. – М.: Издательский Дом «Вильямс», 2006.

Пособия и методические указания

1. *Прут В. В.* Основные комбинаторные алгоритмы на языке С. Алгоритмы на языке С. МФТИ, 2013.
2. *Прут В. В.* Функции алгебры логики. Алгоритмы на языке С. МФТИ, 2013.
3. *Прут В. В.* Полные системы функций алгебры логики. Алгоритмы на языке С. МФТИ, 2013.
4. *Прут В. В.* Сортировка и поиск на языке С. МФТИ, 2014.
5. *Прут В. В.* Введение в графы. Алгоритмы на языке С. МФТИ, 2014.
6. *Прут В. В.* Свойства графов. Алгоритмы на языке С. МФТИ, 2014.
7. *Прут В. В.* Строки на языке С. МФТИ, 2015.
8. *Прут В. В.* Введение. Целые и рациональные числа. МФТИ, 2002.
9. *Прут В. В.* Матрицы. Геометрия. Фракталы. Игры. МФТИ, 2002.
10. *Прут В. В.* Математическая логика. Теория алгоритмов. Рекурсия. Сортировка. Графы. МФТИ, 2002.

Электронные ресурсы:

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

ЗАДАНИЕ 1

(срок сдачи 23–28 октября)

Для каждой задачи в скобках указана ориентировочная трудность по 10-бальной шкале. Преподаватель предлагает задачи в зависимости от уровня курса (основного или продвинутого) и индивидуальной подготовки студента. Итоговая оценка определяется набранными баллами в заданиях и двух контрольных работах. Оценка за задание по 10-бальной шкале определяется как сумма баллов за решенные задачи, деленная на коэффициент 5 – для продвинутых групп и 4 – для основных групп, но не может превышать 9. Оценка 10 баллов выставляется только за индивидуальные особо сложные задачи, предлагаемые преподавателем.

Рекомендуется программы написать и отладить на *Visual Studio* 2013, 2015.

1. (1) Ввести число бит ($\leq 2^{64} - 1$). Вывести число полных эксабайт,

петабайт, терабайт, гигабайт, мегабайт, килобайт, байт, бит.

2. (3) В выделенной области памяти вычислить количество 0 и 1 бит. Найти максимальные последовательности 0 или 1.

3. (1) $n!! \dots! = n(n-k)(n-2k) \dots m$, $m = n \bmod k$, если n не делится на k ; или $m = k$, если n делится на k . Количество ! знаков в обоих случаях равно k .

4. (3) Решить уравнения $ax^2 + bx + c = 0$, $ax^3 + bx^2 + cx + d = 0$, $ax^4 + bx^3 + cx^2 + dx + e = 0$ аналитическими методами при произвольных коэффициентах.

5. (2) Написать функции нахождения корня «произвольной» функции $f(x) = 0$ методом деления пополам и методом Ньютона.

6. (3) Генерировать прямоугольники и рисовать их на экране, который в консольном режиме рисуется графическими символами ASCII: | В3h, — С4h, 7 ВFh, L C0h, J D9h, r DAh и заполняется какими-либо символами. При пересечении прямоугольников общая часть «перекрашивается» другими символами, общая граница удаляется.

7. (4) Написать функцию печати календаря на любой год в стандартном формате.

8. (2) Написать функцию (рекурсивную и нерекурсивную) вычисления значения полинома по алгоритму Горнера и непосредственного вычисления полинома.

9. (4) Написать функцию преобразования целых и/или вещественных чисел из одной системы счисления в другую (для произвольных систем).

10. Предложить и реализовать способ представления чисел в произвольной системе.

11. (4) Написать функцию вычисления ряда с задаваемой точностью

$$\sin z = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} z^{2k+1}, \text{ где } x - \text{вещественное или комплексное}$$

число.

12. (3) Найти максимальное количество (определяемое типом переменных) чисел Фибоначчи различными методами: рекуррентная формула (по определению), непосредственное вычисление, рекурсивная функция, рекурсивная функция с запоминанием, матричная формула.

13. (2) Все числа Фибоначчи выписываются непрерывно: 0 1 1 2 3 5 8 ... Найти в этой последовательности n -ю цифру.

14. (3) Написать функцию (рекурсивную и нерекурсивную) нахождения наибольшего общего делителя по алгоритму Евклида. Вычислить также наименьшее общее кратное.

15. (5) Написать функцию канонического представления натурального числа n в виде произведения натуральных степеней m простых различных чисел: $n = p_1^{\alpha_1} p_2^{\alpha_2} \dots p_m^{\alpha_m}$.

16. (4) Написать функцию, реализующую решето Эратосфена нахождения всех простых чисел $\leq n$.

17. (8) Найти конечную арифметическую прогрессию, состоящую только из простых чисел.

18. (2) Функция вычисляет сумму S делителей числа n (включая 1) и определяет вид числа: совершенного $S = n$, несовершенного $S < n$ и сверхсовершенного $S > n$. Вычислить количество этих чисел $\leq m$.

19. (3) Найти $\sum_p \left(\frac{1}{p} + \frac{1}{p+2} \right)$, где p и $p+2$ - простые числа близнецы.

20. (5) Написать функцию проверки чисел на делимость с помощью известных признаков делимости на 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 17, 19, 20, 23, 25, $10^k - 1$, $10^k + 1$ ($k = 1, 2, 3, \dots$).

21. (3) Написать функцию (рекурсивную и нерекурсивную) вычисления x^k , используя минимальное количество умножений. k - целое, x - целое, вещественное или комплексное число.

22. (4) Вводятся два целых или вещественных числа, между которыми вводится знак (символ) операции (+ - * / ^). Написать одну функцию вычисления таких выражений и вычислить его. Другой вариант. Эти числа и знак вводятся в виде одной строки. Также написать функцию разбора и вычисления такого выражения.

23. (5) Написать функции «рациональной» арифметики: вычисления (+ - * /) в виде несократимой дроби. Дробь (правильная либо нет) есть С-структура. Написать функцию печати рационального числа в

стандартном виде: $\frac{a}{b}$. Ввести два рациональных числа и символ операции, вычислить и напечатать результат.

24. (2) Написать функцию решения системы n линейных уравнений методом Гаусса, где n - один из параметров функции.

25. (2) Функция вычисляет след вещественной динамической матрицы $Tr A = \sum_{i=k} a_{ik}$.

26. (3) Функции вычисляют нормы вещественных динамических матриц:

$$\left(\sum_{i,k=1}^n a_{ik}^2 \right)^{1/2}, \max_i \sum_{k=1}^n |a_{ik}|, \min_i \max_k |a_{ik}|, n \times \max_{ik} |a_{ik}|.$$

27. (3) Написать функцию умножения вещественных динамических матриц $C_{mn} = A_{mq} B_{qn}$ с произвольными параметрами m, n, q .

28. (6) Написать функцию вычисления e^A , где A - квадратная матрица, с заданной точностью.

29. (6) Написать функцию печати всех перестановок чисел $1, 2, \dots, n$ по одному разу.

30. (6) Перечислить все разбиения натурального числа на натуральные слагаемые. В разбиениях слагаемые идут в невозрастающем или неубывающем порядке. Разбиения перечисляются в лексикографическом или обратном лексикографическом порядке. Пример. $n = 4$: $1+1+1+1$, $1+1+2$, $1+3$, $2+2$, 4 .

31. (6) Написать функцию вычисления и печати коэффициентов ряда

$$\arcsin z = \sum_{k=0}^{\infty} \frac{(2k)!}{2^{2k} (k!)^2 (2k+1)} z^{2k+1} = z + \frac{1}{6} z^3 + \frac{3}{40} z^5 + \dots \quad \text{в виде}$$

рациональных чисел. Количество коэффициентов должно быть максимально.

32. (6) Рекуррентные соотношения для чисел Бернулли:

$$\sum_{k=0}^{n-1} C_n^k B_k = 0, \quad \text{или} \quad B_n = \frac{-1}{n+1} \sum_{k=1}^n C_{n+1}^{k+1} B_{n-k}, \quad n \geq 2. \quad \text{Первые}$$

числа есть $B_0 = 1$, $B_1 = -\frac{1}{2}$, $B_2 = \frac{1}{6}$, $B_4 = -\frac{1}{30}$, ...; однако, нечетные

$B_{2n+1} = 0$ при $n \geq 1$, т.е. все, кроме B_1 , а знаки чисел Бернулли с четными номерами чередуются. Написать функцию вычисления B_{2n} , $n = 1, 2, \dots$ и печати в виде рациональных чисел.

33. (7) Написать функции сложения, вычитания, умножения и деления комплексных чисел и кватернионов в численном и символьном виде.

Кватернион $q = a + ib + jc + kd$, где числа i, j, k удовлетворяют

соотношениям $i^2 = j^2 = k^2 = -1$, $ij = -ji = k$, $jk = -kj = i$,
 $ki = -ik = j$ (в частности, $ijk = -1$).

34. (6) Написать функцию нахождения различных типов рифм в словаре русского языка. В простых рифмах совпадают гласные, и, быть может, согласные буквы. В полноценных рифмах совпадают последние ударные гласные звуки (а, о, э, у, ы, и) рифмующихся строк - на первом, втором, третьем, четвертом от конца слоге.

35. (5) Найти в словаре все слова, отличающиеся лишь одной буквой.

36. (6) Найти в тексте частоту символов, слов, слогов, строк, предложений, абзацев, страниц.

37. (5) Написать функции бинарного поиска в массиве чисел, строк.

38. (4) Написать функции шифрования и дешифрования текста простым и усовершенствованным шифром Цезаря. Простой шифр: циклический сдвиг символа на определенное количество позиций ($>$ или <0), которое является ключом. Улучшенный шифр: замещающий алфавит произвольным (случайным) образом определяет соответствие символов.

39. (5) Минимальное покрытие. Сгенерировать множество отрезков прямой с целочисленными координатами концов $[-n, n]$. Выбрать минимальное подмножество, полностью покрывающее отрезок $[-m, m]$, $m < n$. Если такое покрытие при сгенерированном количестве отрезков и их величинами невозможно, сгенерировать дополнительное множество отрезков (по одному или серий).

ЗАДАНИЕ 2

(срок сдачи 4–9 декабря)

1. (3) Написать функцию печати таблицы истинности от n переменных.

2. (4) Написать функцию, которая определяет алгебраические свойства функции алгебры логики. В частности, находила бы, какие элементарные функции логики коммутативны, ассоциативны или дистрибутивны (слева и справа).

3. (8) Написать функцию нахождения всех базисов функций алгебры логики, состоящих из одной или нескольких элементарных функций, исходя из теоремы Поста.

4. (8) Написать функцию, которая выражает все элементарные функции алгебры логики через найденные базисы предыдущей задачи.

5. (4) Написать функцию, которая бы по таблице истинности возвращала бы дизъюнктивную (конъюнктивную) нормальную форму (ДНФ, КНФ).

6. (8) Написать функцию, которая бы по произвольной дизъюнктивной (конъюнктивной) нормальной форме возвращала бы её совершенную дизъюнктивную (конъюнктивную) нормальную форму (СДНФ, СКНФ).
7. (8) Написать функцию, которая по формуле алгебры логики возвращает её представление в виде полинома Жегалкина.
8. (9) В произведении прямоугольных матриц $A_1 \times A_2 \times \dots \times A_n$, где размер матрицы A_i есть m_{i-1}, m_i , расставить скобки так, чтобы вычисление произведения требовало наименьшего числа действий (умножений). Затем сгенерировать матрицы, провести вычисления и сравнить время вычислений.
9. (7) Для n -значных чисел в системе счисления с основанием P вычислить количество чисел, которые не содержат двух подряд идущих одинаковых цифр.
10. (7) Рассматриваются все лексикографически упорядоченные последовательности длиной n , состоящие только из 0 и 1, в которых нет подряд идущих 0. Найти последовательность, находящуюся на m -м месте.
11. (7) Функция определяет правильность представления вещественного числа в формате языка C, заданного строкой.
12. (4) Обойти конем «шахматную» доску $m \times n$ (алгоритм с возвратом). Реализовать эвристический способ ускорения: конь ходит на одну из менее доступных клеток, из которых возможно минимальное количество ходов. В случае нескольких клеток с равными возможностями доступа конь может перемещаться в любую из этих клеток. Усложнение задачи – конечная клетка маршрута задается. Определить, при каких параметрах $m \times n$ не все клетки можно обойти.
13. (4) Сгенерировать (или ввести) последовательность из n различных скобок: $()$, $[]$, $\{\}$, $\langle \rangle$ и определить её корректность.
14. (5) Написать рекурсивные и нерекурсивные функции поиска в глубину и ширину в дереве.
15. (5) Написать функции вычисления высоты дерева, количества узлов, количества листьев, количества внутренних узлов.
16. (5) Написать функцию, которая для бинарного дерева находит расстояние между заданными вершинами.
17. (8) Написать рекурсивные и итеративные функции взаимного преобразования инфиксной, постфиксной и префиксной форм представления арифметического и/или логического выражений.
18. (4) Написать функции вычисления постфиксного и/или префиксного выражения предыдущей задачи.

19. (9) Написать функции построения дерева выражения (синтаксического анализа) из постфиксной формы и обратного преобразования дерева в инфиксную форму.
20. (5) Написать функции печати дерева выражения в «классическом» виде (корень вверху и посередине).
21. (4) Сгенерировать множество целых чисел и представить их как дерево двоичного поиска. Из этого дерева исключить четные числа, сохранив структуру дерева, а затем упорядочить полученное множество.
22. (8) Написать функции, реализующие операции словаря структур: вставить, найти, удалить, найти k -й по величине элемент, сортировать, слить два словаря и др. Словарь организован на основе связанного списка и/или дерева бинарного поиска (BST).
23. (5) Написать функции определения идеальной или AVL сбалансированности BST-дерева.
24. (3) Написать функции сортировки массива чисел различными методами: простым выбором, простыми вставками, простым обменом (метод «пузырька»), «шейкер»–сортировку.
25. (9) Написать функции сортировки массива чисел методом Ноаре («быстрая» сортировка), методом слияния, пирамидальным методом, методом бинарного дерева. Затем сгенерировать случайные последовательности чисел, получить численные результаты для зависимости времени сортировки от длины массива, длины элементов, степени упорядочивания, параметров методов и сравнить их с аналитическими зависимостями и между собой.
26. (5) Написать функцию сортировки строк в лексикографическом порядке.
27. (6) Сгенерировать динамический массив и список (односвязный или двусвязный) одинаковой длины и упорядочить их различными методами. Сравнить скорости сортировки в тактах компьютера с помощью команды *rdtsc* (функции `__rdtsc()`).
28. (9) Сгенерировать полином от $n \geq 1$ переменных, представленный связным списком, и написать функции $+$, $-$, $*$, $/$, упрощения, приведения подобных членов, сортировки сначала по суммам степеней, а затем по степеням переменных.
29. (7) Написать функции генерирования случайного «простого» графа, орграфа, взвешенного графа с n вершинами и m ребрами.
30. (7) Написать рекурсивные и нерекурсивные функции поиска в глубину и ширину в графе (орграфе).
31. (8) Написать функции нахождения эйлера и гамильтонова циклов графа.

32. (8) Написать функции нахождения кратчайших путей в графе (всех, из одной вершины до всех).
33. (9) Написать функции раскраски вершин графа в минимальное количество цветов.
34. (8) Задача Иосифа Флавия: n воинов, стоящих по кругу, убивают каждого m -го. Вычислить номера начальных позиций выбывающих воинов. Рассмотреть различные методы решения и оценить трудоемкость задачи.
35. (8) Сгенерировать координаты вершин многоугольника с прямыми углами, не обязательно выпуклого, но без самопересечений (например, крест). Сначала перемешать координаты случайным образом, а затем отсортировать их. Далее вычислить площадь многоугольника и нарисовать его.
36. (9) Создать хеш-таблицы с целыми и строковыми ключами. Ввести различные хеш-функции и методы разрешения коллизий для открытого и закрытого хеширования. Вычислить время поиска для различных параметров таблицы и выбранных методов.