

Министерство образования Республики Беларусь

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра информатики

А.А. Мелещенко

Практикум

по курсу «Конструирование программ и языки программирования» для
студентов специальности 31 03 04 «Информатика» дневной формы обучения

Минск 2003

УДК 681.3.061 (075.8)

ББК 32.973я73

М 47

Мелешенко А.А.

М 47 Практикум по курсу «Конструирование программ и языки программирования» для студентов специальности 31 03 04 «Информатика» дневной формы обучения / А.А. Мелешенко. – Мн.: БГУИР, 2003. – 38 с.
ISBN 985-444-541-0

Практикум является дополнительным учебным материалом для формирования практических навыков у студентов, изучающих язык программирования С.

УДК 681.3.061 (075.8)

ББК 32.973я73

ISBN 985-444-541-0

© Мелешенко А.А., 2003

© БГУИР, 2003

Содержание

Введение	4
1. Основные конструкции	6
2. Функции	14
3. Массивы и структуры	21
4. Сортировка данных	25
5. Строки	27
6. Динамические структуры данных	33
Литература	37

Библиотека БГУИР

Введение

В разработке программ традиционно выделяют следующие этапы: формализация постановки (перевод задачи на язык математической символики), выбор либо разработка метода решения, разработка алгоритма, кодирование (перевод алгоритма на алгоритмический язык в соответствии с его синтаксисом), тестирование, отладка, защита, сдача в эксплуатацию, сопровождение.

Отладка – процесс выявления, локализации и устранения ошибок в алгоритме и реализующей его программе – осуществляется с помощью тестирования. Синтаксические ошибки с разной степенью эффективности выявляются трансляторами. Наша задача – убедиться в корректности алгоритма, т.е. получить уверенность, что программа выдает результаты, соответствующие задаче и исходным данным. Посредством испытаний (тестирования) либо теоретически невозможно доказать правильность программы; можно лишь спровоцировать появление и устранить в ней как можно больше ошибок.

Тест – совокупность исходных данных для программы вместе с ожидаемыми результатами (с учетом формы представления последних). Готовится не один тест, а их совокупность – *набор тестов*, призванный охватить максимум ситуаций. Испытание программы проводится сразу на всем наборе тестов; если на каком-либо тесте выявилась ошибка, набор тестов нужно «прогонять» заново.

В наборе тестов выделяют три группы:

- «тепличные» - проверяющие программу при корректных, нормальных исходных данных самого простого вида;
- «экстремальные» - на границе области определения в ситуациях, которые могут произойти и на которые нужно корректно реагировать;
- «запредельные» - за границей области определения (а возможно, и здравого смысла) – ситуации, бессмысленные с точки зрения постановки задачи, но которые могут произойти из-за ошибок пользователя или программ, поставляющих исходные данные для тестируемой программы.

К защите допускаются программы:

1. Полностью соответствующие условиям задачи.
2. Не имеющие «замечаний» (warnings) транслятора.

3. Протестированные на «тепличной», «экстремальной» и «запредельной» группах тестов.

Большинство задач имеют не единственное решение. При этом критериями качества программы служат следующие показатели (по убыванию важности):

- оригинальность решения;
- объем памяти, занимаемой программой (с учетом памяти, отводимой под массивы)
- трудоемкость вычислений, т.е. эффективность алгоритма;
- лаконичность и наглядность программы, включая наличие и качество комментариев.

Приветствуется обобщение постановки задачи, т.е. замена частного случая более общим.

Библиотека БГУИР

1. Основные конструкции

Задача А:

1. Временной интервал. Заданы моменты начала и конца некоторого промежутка времени в часах, минутах и секундах (в пределах одних суток). Найти продолжительность этого промежутка в тех же единицах измерения.

2. Коммерция. Коммерсант, имея стартовый капитал k рублей, занялся торговлей, которая ежемесячно увеличивает капитал на $p\%$. Через сколько лет он накопит сумму s , достаточную для покупки собственного магазина?

3. Русские неметрические единицы длины: 1 верста = 500 сажень; 1 сажень = 3 аршина; 1 аршин = 16 вершков; 1 вершок = 44,45 мм. Длина некоторого отрезка составляет p метров. Перевести ее в русскую неметрическую систему.

4. Голодная зима. Суточный рацион коровы составляет u кг сена, v кг силоса и w кг комбикорма. В хозяйстве, содержащем стадо из k голов, осталось s центнеров сена, t тонн силоса и f мешков комбикорма по 50 кг. Сколько еще дней хозяйство сможет кормить коров по полному рациону? Какой из кормов кончится раньше других?

5. Движение без топлива? Владелец автомобиля приобрел новый карбюратор, который экономит 50% топлива, новую систему зажигания, которая экономит 30% топлива, и поршневые кольца, экономящие 20% топлива. Верно ли, что его автомобиль теперь сможет обходиться совсем без топлива? Найти фактическую экономию для произвольно заданных сэкономленных процентов.

6. Цена на молоко. Фермер в начале каждой зимы повышает отпускную цену на молоко на $p\%$, а каждым летом – снижает на столько же процентов. Изменится ли цена на молоко, и если да, то в какую сторону и насколько, через n лет?

7. Переправа. Чапаеву надо под прямым углом к фарватеру преодолеть реку Урал шириной b м. Его скорость в стоячей воде v_1 м/с; скорость течения реки – v_2 м/с. Под каким углом к фарватеру он должен плыть, чтобы его «не снесло»?

Сколько времени займет переправа? Как изменится решение, если посередине реки Чапаева ранили в руку, и его скорость с v_1 м/с упала до v_3 м/с?

8. Шар и ромб. Может ли шар радиусом r пройти через ромбообразное отверстие с диагоналями p и q .

9. Длина высоты. Треугольник ABC задан длинами своих сторон. Найти длину высоты, опущенной из вершины A .

Экстремальные тесты: сумма двух сторон равна третьей; одна из сторон равна нулю.

Шахматы. В следующих задачах позицию каждой шахматной фигуры можно задавать в обычной нотации (например $d7$) или парой чисел, первое из которых задает номер вертикали, а второе – номер горизонтали. При тестировании полезно проверить алгоритм на недопустимых ситуациях, когда несколько фигур стоят на одном поле.

10. Даны натуральные числа k, l, m, n . Требуется выяснить, являются ли поля (k, l) и (m, n) полями одного цвета.

11. Даны натуральные числа k, l, m, n . Требуется выяснить, угрожает ли ферзь, стоящий на поле (k, l) , полю (m, n) .

12. Даны натуральные числа k, l, m, n . Требуется выяснить, угрожает ли конь, стоящий на поле (k, l) , полю (m, n) .

13. На шахматной доске стоят черный король и белые ладья и слон. Проверить, есть ли угроза королю, и если есть, то от кого именно. Учесть возможность защиты (например, ладья не бьет через слона).

14. Время обслуживания. Для каждого посетителя парикмахерской (с одним мастером) известны следующие величины: t – момент его прихода и b – продолжительность его обслуживания. Сколько клиентов обслужит мастер за смену продолжительностью T ? Сколько рабочего времени он потратит на обслуживание?

15. Расписание. Известно время начала и окончания (например 6.00 и 24.00) работы некоторого пригородного автобусного маршрута с одним автобусом на линии, а также протяженность маршрута в минутах (в один конец) и время отдыха на конечных остановках. Составить суточное расписание этого маршрута (моменты отправления с конечных пунктов) без времени на обед и пересменку.

16. Из миль в километры. Получить таблицу пересчета миль в километры и обратно ($1 \text{ миля} = 1,609344 \text{ км}$) для расстояний, не превышающих k км, в следующем виде:

мили	километры
0,6214	1,0000
1,0000	1,6093
1,2428	2,0000
1,8641	3,0000
2,0000	3,2187

17. Расписание звонков. В учебном заведении задается начало учебного дня, продолжительность «пары», продолжительность обычного и большого перерывов (и их «место» в расписании), количество пар. Получить расписание звонков на весь учебный день.

18. Вырубка леса. Леспромхоз ведет заготовку древесины. Первоначальный объем ее на территории леспромхоза составлял p кубометров. Ежегодный прирост составляет $k\%$. Годовой план заготовки – t кубометров. Через сколько лет в бывшем лесу будут расти одни опята?

19. Гуси и кролики. У гусей и кроликов вместе $2n$ лап. Сколько может быть гусей и кроликов (вывести все возможные сочетания)?

20. Из дюймов в метры. Длина отрезка задана в дюймах ($1 \text{ дюйм} = 2,54 \text{ см}$). Перевести значение длины в метрическую систему, т.е. выразить ее в метрах, сантиметрах и миллиметрах. Так, например, $21 \text{ дюйм} = 0 \text{ м } 53 \text{ см } 3,4 \text{ мм}$.

21. Селекция. Селекционер вывел новый сорт зерновой культуры и снял с опытной делянки k кг семян. Посеяв 1 кг семян, можно за сезон собрать p кг семян. Через сколько лет селекционер сможет засеять новой культурой поле площадью s га, если норма высева n кг/га?

22. Русская пирамида. Сколько кругов заданного радиуса r можно вырезать из правильного треугольника со стороной a ?

23. Планировка. Можно ли на прямоугольном участке застройки размером a на b метров разместить два дома размером p на q и r на s метров? Дома можно располагать только параллельно сторонам участка.

24. Текущая стоимость оборудования. Фирма ежегодно на протяжении n лет закупала оборудование стоимостью соответственно s_1, s_2, \dots, s_n рублей в год (эти числа вводятся и обрабатываются последовательно). Ежегодно в результате износа и морального старения (амортизации) все имеющееся оборудование уценивается на $p\%$. Какова общая стоимость накопленного оборудования за n лет?

25. Счастливый год. Введите месяц и день своего рождения. Выясните, какой ближайший год будет для вас счастливым. Год называется счастливым, если остаток от деления суммы его цифр на 10 совпадает с аналогичным остатком сумм цифр месяца или дня рождения.

26. Погашение кредита. Предприниматель, начав дело, взял кредит размером k рублей под $p\%$ годовых и вложил его в свое дело. По прогнозам его дело должно давать прибыль r рублей в год. Сможет ли он накопить сумму, достаточную для погашения кредита, и если да, то через сколько лет?

Задача В:

1. Не используя оператора `if`, присвоить переменной s значение 0, если введенное число x лежит вне отрезков $[2..5]$ и $[-1..1]$, и значение 1 – в противном случае.

2. Дано натуральное число n . Вычислить произведение первых n сомножителей: $1/2 \cdot 7/8 \cdot 13/14 \cdot 19/20 \cdot \dots$

3. Составить программу, которая по двум введенным вещественным числам вычисляет коэффициенты приведенного квадратного уравнения, корнями которого являются эти числа, и печатает это уравнение в виде $x^2+px+q=0$.

4. Составить программу, печатающую ДА или НЕТ в зависимости от того, имеют ли три целых введенных числа одинаковую четность.

5. Целой переменной d присвоить первую цифру из дробной части вещественного положительного числа.

6. Заданное натуральное число n , не превосходящее 1000, записать прописью, например: 375 – «триста семьдесят пять».

7. Для заданного m получить таблицу первых m простых чисел.

8. Перевести заданное целое число в систему римского счета.

Указание. Римские цифры обозначаются следующими латинскими буквами:

1	I	500	D
5	V	1000	M
10	X	5000	V
50	L	10000	X
100	C

9. Десятичное представление заданного натурального числа напечатать *взразрядку*, т.е. вставить пробелы между цифрами.

10. Напечатать столбиком пример на умножение в десятичной системе счисления двух заданных натуральных чисел k и l .

11. Любая целочисленная денежная сумма $s > 7$ р. может быть выдана без сдачи «трешками» и «пятерками». Найти для заданной суммы s необходимое количество «трешек» и «пятерок».

12. Найти 10 первых троек *пифагоровых* чисел, т.е. целых k, l, m таких, что $k^2 + l^2 = m^2$.

13. Найти все натуральные числа, не превосходящие заданного n , десятичная запись которых есть строго возрастающая или строго убывающая последовательность цифр.

14. Каждое из заданных натуральных чисел заменить числом, получающимся при записи его десятичных цифр в обратном порядке.

15. Найти все натуральные числа, не превосходящие заданного m , двоичная запись которых представляет собой симметричную последовательность нулей и единиц (начинающуюся с единицы). Показать десятичную и двоичную записи этих чисел.

16. Найти все натуральные числа от 1 до 1000, которые совпадают с последними разрядами своих квадратов, например: $25^2 = 625$; $76^2 = 5676$.

17. **Тренажер по арифметике.** Пользователь (учитель) вводит разрядность операндов, тип операции: $+$ $-$ $*$ $/$ (на множестве натуральных чисел) и количество примеров. Компьютер генерирует случайным образом операнды, результат операции и выводит ученику серию примеров, в каждом из которых один из операндов или результат «замаскирован», например: $37 * _ = 1591$. Ученик вводит пропущенное число (в приведенном примере – 41); компьютер проверяет правильность и ведет статистику ошибок.

18. Натуральное число называется *совершенным*, если оно равно сумме всех своих простых делителей (например, $6 = 1 + 2 + 3$). Найти все совершенные числа, не превосходящие заданного n .

Примечание. На сегодня известно 24 совершенных числа; все они четные. Вопрос о конечности множества совершенных чисел открыт.

19. Пусть дробные числа (с плавающей точкой во внутреннем представлении), которые сильно отличаются по величине, вводятся с клавиатуры или из файла. Вывести их на экран столбиками, оставляя в числе по 2 цифры после точки, выровняв числа в столбце по правому краю так, чтобы в строке между числами было не менее 1 пробела; ширина столбца определяется самым большим числом в нем.

20. Найти и распечатать все натуральные трехзначные числа, равные сумме кубов своих цифр.

21. Назовем шестизначный автобусный билет удачным, если сумма его цифр делится на 7. Могут ли два билета подряд быть удачными?

22. Определить k -ю цифру последовательности
182764125216343... ,
в которой выписаны подряд кубы натуральных чисел.

23. Найти количество трехзначных чисел, кратных 15, но не кратных 30. Распечатать эти числа.

24. Для заданного e найти наименьшее n такое, что $2^n / n! < e$. Вывести все члены последовательности от 1-го до n -го.

25. Определить k -ю цифру последовательности
112358132134... ,
в которой выписаны подряд все числа Фибоначчи (1, 1, 2, 3, 5, 8,...).

26. Найти все двузначные числа, сумма цифр которых не меняется при умножении числа на 2, 3, 4, 5, 6, 7, 8, 9.

Задачи повышенной сложности:

1. Поменять местами значения целых переменных A и B , не используя дополнительные переменные.

2. Три друга были свидетелями ДТП. Первый заметил, что номер нарушителя делится на 2, 7 и 11. Второй запомнил, что в записи номера имеется всего две

различные цифры, а третий – что сумма цифр равна 30. Определить четырехзначный номер нарушителя.

Замечание. Можно ли определить номер нарушителя, если показания дадут только два друга?

3. Даны треугольник и прямая. Определить, пересекает ли прямая контур треугольника.

Библиотека БГУИР

2. Функции

Задача А:

1. Вычислить факториалы всех чисел от 1 до 100.

Примечание. Число $100!$ не может быть представлено ни в одном стандартном типе переменных и не поместится на одной строке экрана дисплея.

2. Пусть значение функции $f(n)$ равно количеству символов в русской записи количественного числительного n : $f(1) = 4$ (), $f(3) = 3$ (), $f(42) = 9$ и т.д. Найти все натуральные n , для которых $f(n) = n$.

3. Вывести с помощью рекурсии двоичный эквивалент целого десятичного числа.

4. Разработать функцию, которая на основе входного положительного числа n возвращает соответствующее число Фибоначчи.

5. Разработать функцию $Fibonacci()$, где при вычислении чисел Фибоначчи используется рекурсия.

6. Разработать программу, обслуживающую меню выбора числа в русские метрические единицы из меню.

7. Реализуйте функцию $chline(char c, int i, int j)$, которая выводит запрошенный символ i раз в j строках экрана.

8. Создайте функцию, заменяющую содержимое двух переменных их суммой и разностью.

9. Создайте функцию, заменяющую содержимое двух целых переменных результатом возведения первого числа в степень второго и результатом возведения второго числа в степень первого.

10. Гармоническое значение двух чисел получается путем получения обратных значений двух чисел, усреднения их и получения обратного значения

результата. Создайте функцию, которая принимает два аргумента типа *double* и возвращает гармоническое значение двух чисел.

11. Разработать и протестировать рекурсивную функцию *power()*, которая возвращает результат возведения числа *double* в целую степень. Встройте в нее функцию, которая любой степени числа 0 присваивает значение 0, а любому числу в степени 0 присваивает значение 1.

12. Разработать функцию, принимающую 2 аргумента. Функция должна выводить число, которое является представлением первого аргумента в системе счисления с основанием, определяемым вторым аргументом.

13. Написать и протестировать функцию *compress()*, которая «сжимает» строку, удаляя из нее все пробелы.

14. Написать и протестировать функцию для вычисления числа сочетаний по формуле

$$C_n^k = \frac{n!}{k!(n-k)!} = \frac{n(n-1)\dots(n-k+1)}{k!}.$$

15. Написать и протестировать функцию, которая «переворачивает» строку, передаваемую ей в качестве параметра.

16. Натуральное число *m* представить в виде суммы квадратов двух натуральных чисел. Выдать сообщение, если такое представление невозможно.

17. Даны вещественные числа *a*, *b*, *c*, *d*, *e*, *f*. Переменной *s* присвоить значение 1, если оба уравнения $ax^2 + bx + c = 0$ и $dx^2 + ex + f = 0$ имеют вещественные корни и при этом все корни первого уравнения лежат между корнями второго уравнения. В противном случае переменной *s* присвоить значение 0. (Для нахождения корней квадратного уравнения использовать функцию.)

18. Даны натуральные числа *n* и *k*. Определить *k*-ю справа цифру числа *n*.

19. Поле шахматной доски определяется парой натуральных чисел, первое из которых задает номер вертикали, а второе – номер горизонтали. Даны натуральные числа *k*, *l*, *m*, *n*. Требуется, если возможно, с поля (*k*, *l*) одним

ходом коня попасть на поле (m, n) . Если нет, то определить, можно ли это сделать за два хода. В случае успеха указать промежуточное поле. Использовать функцию.

20. Описать функцию $\text{minmax}(x, y)$, которая присваивает первому параметру большее, а второму – меньшее из значений x и y . Используя эту функцию, перераспределить введенные значения переменных A, B, C так, чтобы стало $A \leq B \leq C$.

21. Подсчитать, сколько раз во введенном тексте встречается слово «по». Слова в тексте разделяются пробелами.

22. Даны натуральные числа n, m , и k . Написать и протестировать функцию, которая возвращает суммы, полученные в результате сложения k младших цифр числа n и k старших цифр числа m .

23 Дано натуральное число a . Вычислить сумму k старших (находящихся слева) цифр числа.

24. Написать и протестировать функцию, которая по заданному натуральному числу определяет количество цифр в нем и их сумму.

25. Выяснить, сколько простых чисел находится в интервале $[n, m]$, и распечатать их. Для определения, является ли очередное число простым, составить функцию.

26. По введенному символу установить, в каких позициях его двоичного кода записаны нули.

27. Дано натуральное число n . Распечатать число, которое получится после выписывания цифр числа n в обратном порядке. (Для получения нового числа составить функцию).

28. Во введенном тексте подсчитать количество слов, содержащих три буквы 'е'. Слова разделены пробелами.

29. Написать и протестировать функцию, которая преобразует целое без знака в его восьмеричное символьное представление (библиотечные функции для преобразования числа в троюк и формат вывода «%o» не использовать).

30. Разработать функцию, возвращающую наименьшее общее кратное трех заданных натуральных чисел.

Задача В:

1. По введенному символу установить, в каких позициях его двоичного кода записаны единицы.

2. Написать и протестировать функцию, которая преобразует целое без знака в его двоичное символьное представление (библиотечные функции для преобразования числа в строку не использовать).

3. Написать и протестировать функцию, которая по заданному натуральному числу определяет его первую и последнюю цифры.

4. Написать и протестировать функцию для вычисления площади треугольника, заданного координатами вершин.

5. Из заданного множества точек на плоскости выбрать такие три точки A , B , C , чтобы внутри треугольника ABC содержалось максимальное количество точек этого множества.

6. По заданным целочисленным координатам четырех точек на плоскости определить, какую геометрическую фигуру они образуют, если их соединить в порядке ввода координат точек. Ответы: четырехугольник с самопересечением, четырехугольник, выпуклый четырехугольник, трапеция, параллелограмм, ромб, прямоугольник, квадрат.

7. Написать и протестировать функцию, которая по заданной строке Str формирует новую строку, состоящую только из цифр, входящих в Str .

8. Из заданного по плоскости множества точек выбрать такие три, которые составляют треугольник наибольшей площади.

9. Даны две квадратные матрицы. Напечатать ту из них, которая имеет минимальный «след» (т.е. сумму элементов главной диагонали). Использовать функцию для нахождения следа матрицы и функцию печати матрицы.
10. В массиве найти отрезок максимальной длины, в котором первое число равно последнему, второе – предпоследнему и т.д. Напечатать характеристики этого отрезка (длину и номер первого элемента).
11. Написать и протестировать функцию, которая «переворачивает» строку, передаваемую ей в качестве параметра.
12. Вычислить факториал числа n . Факториал числа представить в виде целочисленного массива десятичных цифр.
13. Написать и протестировать функцию для сложения и вычитания вещественных матриц. Одним из формальных параметров должен быть признак вида операции.
14. Имеется k селений. Если в селении i расположена больница, то поездка в селение j займет время $a[i][j]$. Найти номер селения i , в котором выгоднее всего разместить больницу (суммарное время поездок из i во все другие селения должно быть минимальным).
15. Составить и протестировать функцию для замены символов ‘:’ на ‘.’ в заданной строке, начиная с указанной позиции.
16. Составить обычную и рекурсивную функции для нахождения наибольшего общего делителя двух чисел. Сравнить время работы обеих функций.
17. Написать и протестировать функцию для нахождения в прямоугольной матрице номера строки, имеющей максимальную сумму элементов.
18. Переформировать матрицу таким образом, чтобы ее строки располагались по возрастанию их поэлементных сумм.

19. Написать и протестировать функцию, которая определяет, располагаются ли буквы в заданной символьной строке в алфавитном порядке.
20. Выяснить, сколько различных чисел содержится в заданном одномерном целочисленном массиве.
21. Написать и протестировать функцию, переставляющую в обратном порядке элементы главной диагонали квадратной матрицы.
22. Дан целочисленный массив $a(n)$. Определить три наибольших элемента этого массива.
23. Написать и протестировать функцию, подсчитывающую количество минимальных элементов в целочисленной матрице.
24. Дана целочисленная матрица $A(n, m)$. Заменить нулями элементы матрицы, стоящие на пересечении строк и столбцов, в которых имеется хотя бы по одному нулю.
25. Написать и протестировать функцию, которая находит в массиве минимальный по модулю элемент и заменяет им все элементы с нечетными номерами.
26. Переформировать матрицу таким образом, чтобы ее столбцы располагались по убыванию их поэлементных сумм.
27. Написать и протестировать функцию, которая в прямоугольной матрице находит сумму элементов j -й строки.
28. Найти седловые точки матрицы (седловой точкой называется элемент, являющийся минимальным в строке и максимальным в столбце).
29. Написать и протестировать функцию, которая подсчитывает, сколько раз в заданной строке встретился указанный символ.
30. Написать и протестировать функцию, подсчитывающую количество положительных элементов в массиве.

Задачи повышенной сложности:

1. Написать эффективную функцию для возведения числа в положительную целую степень.

2. Используя метод «Решето Эратосфена», по заданному натуральному $N > 1$ найти все простые числа, меньшие числа N .

Метод заключается в следующем. Выпишем все числа от 2 до N . Первое простое число 2. Вычеркнем все числа, кратные 2. Первое оставшееся число 3 – простое. Вычеркнем все числа, кратные 3 и т.д. В результате останутся только простые числа.

3. Среди заданного на плоскости множества точек найти такую, сумма расстояний от которой до остальных максимальна.

4. Дано натуральное число n . Найти разложение числа на простые множители.

3. Массивы и структуры

1. Определить, является ли данная матрица ортонормированной, т.е. такой, в которой скалярное произведение каждой пары различных строк равно нулю, а скалярное произведение каждой строки на себя равно единице.

2. В двумерном массиве $X(n, m)$ все числа различны. В каждой строке находится минимальный элемент, затем среди этих чисел находится максимальное. Напечатать индексы (номер строки и номер столбца) этого элемента.

3. Найти максимальное из чисел, встречающихся в заданном целочисленном массиве $a(n)$ более одного раза.

4. Ввести структуру для описания *комплексного числа*. Составить и протестировать функции для:

а) преобразования комплексного числа из алгебраической формы в показательную;

б) преобразования комплексного числа из показательной формы в алгебраическую;

в) получения сопряженного комплексного числа;

г) возведения комплексного числа в целую положительную степень;

д) умножения комплексных чисел в алгебраической форме;

е) умножения комплексных чисел в показательной форме;

ж) деления комплексных чисел в показательной форме;

з) деления комплексных чисел в алгебраической форме.

5. Массив структур содержит информацию о студентах группы: в первом поле стоит фамилия, во втором – возраст, в третьем – рост, в четвертом – средний балл в сессию и т.д. (i -й элемент массива описывает i -го студента).

Студент называется *среднестатистическим* по k -му параметру, если на нем достигается *минимум модуля разности* среднего арифметического чисел k -го столбца и значения k -го параметра этого студента. Аналогично определяется *уникальный* по k -му параметру студент (на нем достигается *максимум*).

Студент называется *самым средним*, если он является среднестатистическим по самому большому количеству параметров. Аналогично определяется *самый уникальный* студент.

Выяснить, кто в группе является:

- а) самым средним,
- б) самым уникальным,
- в) самым средним среди самых уникальных,
- г) самым уникальным среди самых средних.

6. Ввести структуру для описания понятия *алгебраический полином*. Составить и протестировать функции для:

- а) ввода полинома;
- б) вывода полинома;
- в) нормализации полинома;
- г) сложения полиномов;
- д) вычитания полиномов;
- е) умножения полиномов;
- ж) деления полиномов;
- з) дифференцирования полинома;
- и) интегрирования полинома.

7. Ввести перечислимые типы *масть, достоинство*. С их помощью описать как структуру переменную *карта*. Составить и протестировать функцию

БЪЕТ (K1, K2, KM),

которая проверяет, бьет ли K1 карту K2, с учетом того, что масть KM является козырной.

8. Описать как структуру переменную *время* (с полями часы, минуты, секунды). Составить и протестировать функции:

а) СЛЕД_СЕК(t, t1, d)

б) ИНТЕРВАЛ (t1, t2, d),

которые вычисляют время d, прошедшее от времени t1 до времени t2.

9. Ввести перечислимые типы *вертикаль, горизонталь* для обозначения клеток шахматной доски. Составить и протестировать функции:

а) ХОД_ФЕРЗЯ (K1, K2),

которая проверяет, может ли ферзь за один ход перейти с поля K1 на поле K2;

б) ХОД_КОНЯ (K1, K2),

которая вычисляет, за сколько ходов конь может перейти с поля K1 на поле K2.

10. Ввести структуру (с полями *числитель* и *знаменатель*) для описания понятия *рациональное число*. Составить и протестировать функции:

а) РАВНО (A, B),

которая проверяет, равны ли друг другу рациональные числа A, B;

б) МАКС (X, N),

которая возвращает наибольшее из массива X[N] рациональных чисел;

в) СЛОЖ (A, B, C),

которая записывает в C результат сложения рациональных чисел A и B;

г) МИН (A, B),

которая возвращает наименьшее из двух рациональных чисел A и B;

д) УМН (A, B, C),

которая записывает в C результат перемножения рациональных чисел A и B.

11. Ввести структуру (с полями *число*, *месяц*, *год*) для описания понятия *дата*. Составить и протестировать функцию, которая:

а) вычисляет интервал (в днях), прошедший между двумя датами;

б) по порядковому номеру дня в году определяет число и месяц года, соответствующие этому дню;

в) по введенной дате распечатывает дату на N дней вперед.

12. Определить структуры, описывающие *шар* и *точку* в трехмерном пространстве. Составить и протестировать функцию, которая проверяет, находится ли точка внутри заданного шара.

13. Ввести структуру для регистрации автомашин. Она должна иметь следующие поля:

дату регистрации (структура с полями – день, месяц, год);

марку машины;

год выпуска;

цвет;

номер.

Написать и протестировать функции:

регистрация новой машины;

удаление машины из регистрационного списка;

поиск машины по любой из комбинаций признаков.

14. В доме N этажей и три лифта. Каждый лифт может быть свободным или занятым. Человек стоит на одном из этажей и собирается вызвать либо ближайший свободный лифт, либо ближайший занятый, направляющийся в сторону этажа, где находится человек.

Распечатать начальную конфигурацию (расстановку, занятость и направление движения лифтов, местоположение человека), а также номер лифта, который будет вызван.

Использовать функции ВВОД, ВЫВОД, ВЫБОР_ЛИФТА.

15. Пусть ЭВМ не умеет работать с вещественными числами, а имеет только операции и функции для работы с символами, строками и целыми числами.

Реализовать функции для:

а) ввода;

б) вывода;

в) сложения;

г) вычитания;

д) умножения

вещественных чисел. (Числа вводятся как строки, разделяются на целую и дробную части и над ними, как над целыми числами, с учетом межразрядных переносов, выполняются операции.)

4. Сортировка данных

1. Написать и протестировать функции сортировки записей и поиска их по ключам для следующих методов:

- 1) линейный выбор с обменом, бинарный поиск;
- 2) челночная сортировка, бинарный поиск;
- 3) сортировка Шелла, бинарный поиск;
- 4) быстрая сортировка, бинарный поиск;
- 5) стандартный обмен, интерполяционный поиск;
- 6) линейная вставка, интерполяционный поиск;
- 7) центрированная вставка, бинарный поиск.

Запись имеет три поля, например, фамилия, имя, номер телефона. Иметь не менее 30 записей. Поиск – по любому ключу, задаваемому из меню.

2. Написать и протестировать функции сортировки целочисленных массивов и поиска ключей в них по следующим методам:

- 1) челночная сортировка, бинарный поиск;
- 2) сортировка Шелла, бинарный поиск (рекурсивная функция);
- 3) быстрая сортировка, бинарный поиск;
- 4) центрированная вставка, интерполяционный поиск.

Тест сортировки: сортировка целочисленного массива размером n , элементы которого – случайные величины, распределенные в интервале $(0, n-1)$.

Тест поиска: поиск m элементов в отсортированном массиве.

3. Исследовать возможности адаптации различных методов сортировки к структуре исходного массива. С этой целью определить время сортировки целочисленного массива объемом n для следующих вариантов представления исходного массива:

- неупорядоченный,
- почти упорядоченный,
- упорядоченный в противоположном направлении.

Методы, подлежащие исследованию:

- 1) линейный выбор с обменом, центрированная вставка;
- 2) челночная сортировка, линейная вставка;
- 3) сортировка Шелла, линейная вставка;
- 4) быстрая сортировка, бинарная вставка;
- 5) стандартный обмен, бинарная вставка.

4. Провести сравнительный анализ эффективности следующих методов сортировки:

- 1) линейный выбор с обменом, челночная сортировка, двоичная вставка;
- 2) сортировка Шелла, центрированная вставка;
- 3) стандартный обмен, быстрая сортировка, линейная вставка.

Предлагаемый тест: сортировка целочисленного массива размером n , элементы которого – случайные величины, распределенные в интервале $(0, n-1)$.

5. Провести сравнительный анализ эффективности не менее двух вариантов быстрой сортировки целочисленных массивов большого размера ($n > 5000$).

6. Написать и протестировать функцию челночной сортировки целочисленного массива с управляемым направлением (возрастание/убывание) сортировки по признаку.

Заголовок функции должен иметь вид

```
void shatl(int* a, int n, char* pr);
```

где a – сортируемый массив;

n – размер массива;

pr – признак, управляющий направлением сортировки;

$pr = "incr"$ – сортировка по возрастанию;

$pr = "decr"$ – сортировка по убыванию.

7. Провести сравнительный анализ эффективности методов сортировки вставками: линейной, двоичной, центрированной.

Предлагаемый тест: сортировка целочисленного массива размером n , элементы которого – случайные величины, распределенные в интервале $(0, n-1)$.

5. Строки

Задача А:

1. Написать и протестировать функцию, которая преобразует строку восьмеричных цифр в эквивалентное ей целое десятичное число.
2. Написать и протестировать функцию, которая по заданной строке `Str` формирует новую строку, состоящую только из цифр, входящих в `Str`.
3. Зашифровать текст следующим образом: записать его в матрицу по строкам, а затем переписать по спирали от центра. Прочитать зашифрованный текст.
4. Даны две символьные строки, состоящие только из цифр (длина каждой – более 10 символов). Считая, что в этих строках находятся очень длинные целые числа, сформировать третью строку – сумму этих чисел.
5. Написать и протестировать функцию, которая в строке, передаваемой ей в качестве параметра, заменяет каждый второй элемент на заданный символ.
6. Ввести предложение, слова в котором разделены пробелами и запятыми. Распечатать это предложение, удалив из него те слова, которые встретились там более одного раза.
7. Написать и протестировать функцию, возвращающую номер самого правого вхождения заданного символа во введенную строку. Если символ не входит в строку, должно возвращаться -1 .
8. Дан произвольный текст. Отредактировать его так, чтобы:
 - а) между словами был ровно один пробел;
 - б) предложения в тексте разделялись ровно двумя пробелами.
9. Написать и протестировать функцию, которая определяет, является ли симметричной заданная символьная строка.
10. Ввести два предложения и распечатать самые длинные слова, общие для этих предложений. Если нужных слов нет – сообщить об этом.

11. Написать и протестировать функцию, которая определяет, совпадают ли в заданной строке первая и последняя буквы.
12. Ввести предложение, слова в котором разделены пробелами и запятыми. Распечатать те слова, которые являются обращениями других слов в этом предложении. Если нужных слов нет – сообщить об этом.
13. Написать и протестировать функцию, которая по заданной строке *Str* формирует новую строку, состоящую только из несовпадающих цифр, входящих в *Str*.
14. Из введенного текста распечатать все слова наименьшей длины.
15. Распечатать введенную строку, удалив из нее символы, не являющиеся буквами и цифрами, и заменив каждую цифру на **.
16. Написать и протестировать функцию *ISSUBSTR(str1, str2)*, которая выясняет, является ли строка *str1* подстрокой строки *str2*. Функция должна возвращать номер позиции, с которой начинается подстрока, либо -1 , если подстрока не найдена.
17. Во введенном тексте подсчитать количество слов, считая словом последовательность букв и цифр, начинающуюся с буквы. (Слова разделяются пробелами).
18. Написать и протестировать рекурсивную функцию *STOI(n, str)*, которая преобразует строку десятичных цифр в целое число.
19. Во введенном тексте подсчитать количество символов в слове максимальной длины. (Слова разделяются пробелами).
20. Ввести произвольный текст. Вычислить среднее число слов в предложении и среднюю длину предложения.
21. Написать и протестировать функцию, которая «сжимает» строку, удаляя из нее каждый второй элемент.

22. Проверить, имеется ли в заданном тексте баланс открывающих и закрывающих круглых скобок.

23. Написать и протестировать функцию *compress()*, которая «сжимает» строку, удаляя из нее все пробелы.

24. Написать и протестировать аналог функции *STRNCMP()*.

25. Выяснить, есть ли во введенном тексте слова, начинающиеся с буквы A, и сколько таких слов.

26. Написать и протестировать аналог функции *STRNCAT()*.

27. Выяснить, есть ли во введенном тексте слова, оканчивающиеся на 'r', и сколько таких слов. (Слова разделяются пробелами).

28. Вывести вертикальную гистограмму длин слов введенного текста.

29. Среди цифр введенной строки распечатать ту, которая появлялась чаще других. Если таких цифр было несколько, распечатать ту, что встретилась первой.

30. Дан произвольный текст. Напечатать в алфавитном порядке все буквы, которые входят в этот текст по одному разу.

Задача В:

1. Зашифровать введенный текст, заменив каждый символ на символ, стоящий через один от данного в таблице кодировки. Исходное разбиение на строки должно быть сохранено.

2. Написать и протестировать рекурсивную функцию *REVERSE(str)*, которая переворачивает данную строку на том же самом месте. Сравнить время ее работы и время работы нерекурсивной версии.

3. Написать и протестировать функцию, которая преобразует строку двоичных цифр в эквивалентное ей целое десятичное число.
4. Написать и протестировать функцию $I_TO_B(n, s, b)$, которая переводит целое число n в строку s , представляющую число в системе счисления с основанием b .
5. Составить и протестировать функцию для замены символов ‘.’ на ‘.’ в заданной строке, начиная с указанной позиции.
6. Написать и протестировать функцию $DELETE(s1, s2)$, которая удаляет из строки $s1$ все символы, встречающиеся в строке $s2$.
7. Написать и протестировать функцию, которая определяет, располагаются ли буквы в заданной символьной строке в алфавитном порядке.
8. Написать и протестировать функцию $ESCAPE(str1, str2)$, которая при копировании текста из $str1$ в $str2$ преобразует литеры «новая строка» и «табуляция» в видимые последовательности литер $\backslash n$ и $\backslash t$. Сделать также функцию, выполняющую обратное преобразование.
9. Написать и протестировать функцию, которая определяет, входит ли каждая буква в заданную строку не более двух раз.
10. Составить программу, которая реверсирует каждое слово строки str .
11. Написать и протестировать функцию, которая преобразует строку шестнадцатеричных цифр в эквивалентное ей целое десятичное число.
12. Ввести строку, состоящую только из цифр и букв. Распечатать те группы цифр, в которых цифра 7 встречается не более двух раз. (Группа цифр – это последовательность цифр, обрамленная буквами).
13. Написать и протестировать функцию, которая определяет, входит ли каждая буква в заданную строку не менее двух раз.

14. Распечатать строку, которая получается из введенной строки следующим образом: каждая цифра заменяется на заключенную в круглые скобки последовательность литер '+' (если цифра четная) или '-' (если цифра нечетная), длина которой равна числу, изображаемому цифрой.

15. Ввести текст, состоящий только из цифр и букв. Выяснить, верно ли, что сумма числовых значений цифр, находящихся в тексте, равна длине текста.

16. Распечатать введенную строку, исключив из нее те символы, которые находятся между скобками '(' ')'. Сами скобки не удалять. Если хотя бы одной скобки нет – сообщить об этом.

17. Распечатать введенную строку, заменив строчные буквы прописными и повторив дважды каждую цифру.

18. Распечатать те слова, в которых либо буквы упорядочены по алфавиту, либо каждая буква входит в слово не менее двух раз (т.е. слова типа *ABBA*, *BEER*).

19. Определить, сколько слов во введенном тексте начинаются и оканчиваются одной и той же буквой. (Слова разделены пробелами).

20. Составить частотный словарь вводимого текста. Распечатать его по алфавиту, а справа от каждого слова – частоту, с которой оно встретилось.

21. Написать и протестировать функцию, которая «переворачивает» строку, передаваемую ей в качестве параметра.

22. Определить, является ли введенная строка правильной записью целого десятичного числа без знака.

23. Во введенной строке подсчитать наибольшее количество одинаковых букв, идущих подряд.

24. Написать и протестировать аналог функции *STRNCPY()*.

25. Подсчитать, сколько раз во введенном тексте встречается слово "ok". (Слова в тексте разделены пробелами).

26. Распечатать в порядке, обратном алфавитному, все буквы, которые входят в текст не менее трех раз.

27. Написать и протестировать функцию, которая подсчитывает, сколько раз в заданной строке встретился указанный символ.

28. Среди введенных слов распечатать сначала те, которые начинаются и оканчиваются одной и той же буквой, а затем – все остальные (слова выводить по одному на строке).

29. В последовательности введенных символов (последний '\$') определить порядковый номер первой буквы R (с учетом верхнего/нижнего регистров).

30. Распечатать введенное предложение, удалив из него слова, которые состоят менее чем из трех букв.

Задачи повышенной сложности:

1. Построить вертикальную гистограмму числа вхождений каждой строчной латинской буквы в текст.

2. Произвести выравнивание по правому краю введенного текста, для чего к каждой строке применить функцию $WIDE(str, k)$, которая равномерно вставляет пробелы между словами так, чтобы длина строки str стала равной k . (Величина k должна быть больше длины самой длинной строки текста).

3. Дано целочисленное арифметическое выражение, записанное как строка, в десятичной системе счисления. Проверить правильность записи и вычислить значение этого выражения. Выражение записывается без скобок. Операции выполняются в порядке их следования.

6. Динамические структуры данных

1. Написать и протестировать функции поиска, включения и исключения элемента бинарного дерева с заданным ключом.

2. Определить станцию пересадки в московском метро, если задаются начальная и конечная станции маршрута. Итоговый путь должен отнимать минимальное время (считать, что на пересадку тратится 5 мин, а на проезд между двумя станциями – 2 мин).

3. Многочлен от одной переменной X можно представить как связанный список с узлами вида:

(коэффициент при X , степень X , ссылка на следующий узел)

Используя список:

а) по введенной безошибочной записи многочлена построить его представление в виде списка;

б) сложить два многочлена, представленных в виде списка (нулевые слагаемые исключить из результирующего списка);

в) проверить на равенство два многочлена;

г) вычислить значение многочлена $s(x)$, представленного в виде списка, в целочисленной точке X ;

д) по многочлену $s(x)$ построить его производную – многочлен $p(x)$;

е) привести подобные члены в многочлене и расположить его по убыванию степеней X ;

ж) перемножить два многочлена, заданных в виде списков;

з) распечатать многочлен, заданный в виде списка, в обычном виде (например так: $52y^3 - 6y^2 + y$).

4. Известно, что заданный граф – не дерево. Выяснить, можно ли из него удалить одну вершину (вместе с инцидентными ребрами) так, чтобы в результате получилось дерево.

5. Составить программу, формирующую «перекрестные ссылки», т.е. печатающую список слов, которые встречаются в анализируемом файле, а для каждого слова – список номеров строк, в которых это слово встречается. При решении задачи рекомендуется использовать следующие структуры данных:

```
struct LIST //список номеров строк для данного слова
{
    int num;
    struct LIST *p;
}
```

```
struct NODE //узел дерева с информацией об очередном слове
{
    char *word;
    struct LIST *lines;
    struct NODE *left;
    struct NODE *right;
}
```

6. Мостом графа называется такое ребро, удаление которого приводит к увеличению числа компонент связности графа. Найти все мосты в заданном графе.
7. Используя стек, написать нерекурсивную версию алгоритма Хоара быстрой сортировки.
8. Определить изоморфны ли два заданных графа.
9. Два бинарных дерева подобны, если либо оба они пусты, либо оба непусты, и при этом их левые и правые поддеревья подобны. Определить, являются ли два дерева подобными.
10. Изобразить графически заданный планарный граф так, чтобы его ребра не пересекались.
11. Два бинарных дерева зеркально подобны, если они либо оба пусты, либо оба непусты, и при этом левое поддерево одного из них подобно правому поддереву другого и наоборот. Определить, являются ли два дерева зеркально подобными.
12. Определить, изоморфен ли заданный граф своему дополнению.

Для организации вычисления значения выражения на ЭВМ удобнее вместо обычной (инфиксной) записи использовать постфиксную (или польскую инверсную) запись – ПОЛИЗ. При вычислении выражения, записанного в ПОЛИЗе, операции выполняются в том порядке, в котором они встречаются при просмотре выражения слева направо; поэтому отпадает необходимость

использования скобок и многократного сканирования выражения из-за различного старшинства операций. Например, выражению $2+3*4$ соответствует ПОЛИЗ $234*+$, а выражению $(a+(b^c)^d)*(c+f/d)$ – запись $abc^d^+efd/+*$.

Используя стек:

13. Вычислить как целое число значение выражения, записанного в ПОЛИЗе
14. Выражение, записанное в ПОЛИЗе, перевести в инфиксную форму и распечатать.
15. По введенной формуле построить ее ПОЛИЗ (при записи формулы используются только операции $+$, $-$, $*$, $/$ и буквы – операнды)
16. В заданном графе найти все его подграфы, которые являются ободами.
17. Найти диаметр графа, т.е. максимум расстояний между всевозможными парами его вершин.
18. Найти медиану графа, т.е. такую его вершину, что сумма расстояний от нее до остальных вершин минимальна.
19. Написать программу сложения двух длинных целых чисел, представленных в виде строк, используя:
 - а) круговые скобки;
 - б) двунаправленные списки.
20. В заданном графе найти максимальный по количеству вершин полный подграф (клика).
21. Закодировать введенное сообщение с использованием алгоритма Хаффмена.
22. Выяснить, имеется ли в заданном графе эйлеров путь или эйлеров цикл, и найти эйлеров цикл (если он есть).
23. Распечатать слова текста, отсортированные в порядке убывания частоты их встречаемости (рядом с каждым словом выводить значение счетчика частоты его вхождений в текст). При решении задачи использовать дерево следующей структуры:

```
struct NODE
```

```
{  
    char *word;  
    int k;  
    struct NODE *left;  
    struct NODE *right;  
}
```

24. Выяснить, имеется ли в заданном графе гамильтонов путь или гамильтонов цикл, и найти гамильтонов цикл (если он есть).

25. Написать и протестировать функции для включения, исключения и поиска элемента кругового списка для:

а) списка без заголовка;

б) списка с заголовком (заголовок может содержать некоторую информацию о списке, например, число элементов в списке).

26. В системе двухсторонних дорог для каждой пары городов указать длину кратчайшего пути между ними.

27. В файле находится текст, в котором использованы скобки трех типов: (), [], {}. Используя стек, проверить, соблюден ли баланс скобок в тексте.

28. Заданная система двухсторонних дорог такова, что для любой пары городов можно указать соединяющий их путь. Найти такой город, сумма расстояний от которого до остальных городов минимальна.

29. Написать и протестировать функции включения, удаления и чтения очередного элемента стека объемом n элементов. В случае невозможности включения (переполнения стека), удаления из пустого – стека выставлять флаг.

30. В системе двухсторонних дорог за проезд каждой дороги взимается некоторая пошлина. Найти путь из города A в город B с минимальной величиной $S + P$, где S – сумма длин дорог пути, а P – сумма пошлин проезжаемых дорог.

Литература

1. Прата С. Язык программирования С. Лекции и упражнения. Учебник: Пер. с англ. СПб.: ООО «ДиаСофтЮП», 2002.
2. Крячков А. Программирование на С и С++. Практикум: Учеб. пособие для вузов. – М.: Горячая линия – Телеком, 2000.
3. Юркин А. Задачник по программированию. – СПб.: Питер, 2002.

Библиотека БГУИР

Учебное издание

Мелешенко Александр Александрович

Практикум

по курсу «Конструирование программ и язык программирования» для студентов специальности 31 03 04 «Информатика» дневной формы обучения

Редактор Е.Н. Батурчик
Компьютерная верстка Е.Г. Реут

Подписано в печать 11.08.2003.

Формат 60x84 1/16.

Бумага офсетная. Печать ризографическая.

Гарнитура Times.

Усл.печ.л. 2,44. Уч.-изд. л. 1,5. Тираж 100 экз.

Заказ 285.

Издатель и полиграфическое исполнение:

Учреждение образования

“Белорусский государственный университет информатики и радиоэлектроники”

Лицензия ЛП № 156 от 30.12.2002

Лицензия ЛВ № 509 от 03.08.2001

220013, Минск, П. Бровки, 6