

**Областная олимпиада по специальности
09.02.07 «Информационные системы и программирование».**

Тренировочные задачи для подготовки к решению профессионального задания региональной олимпиады (2020).

Тренировочные задачи можно условно разбить на следующие темы:

1. Задачи на знание и сообразительность. Перебор.
2. Поиск в ширину. Поиск в глубину.
3. Динамическое программирование.
4. Алгоритмы на графах. Минимальное остовное дерево.
5. Алгоритмы на графах. Кратчайший маршрут.

К задачам прилагается литература с решением.

Типовые задачи по темам

1. Задачи на сообразительность. Перебор.

Задача 1. Даны N целых чисел X_1, X_2, \dots, X_N . Расставить между ними знаки «+» и «-» так, чтобы значение получившегося выражения было равно заданному S .

Ограничения: $2 < N < 24, 0 < X_i < 50000$.

Ввод из файла in.txt. В первой строке находятся числа N и S . В следующей строке - N чисел через пробел.

Вывод в файл out.txt. Если получить требуемый результат невозможно, вывести «No solution», если можно – вывести равенство. Если решение не единственное, вывести любое.

Примеры

Пример 1	Пример 2
Ввод: 3 10 15 25 30	Ввод: 2 100 10 10
Вывод: 15+25-30=10	Вывод: No solution

Решение см. [1] стр. 26, 94-95.

Задача 2. Даны n целых чисел X_1, X_2, \dots, X_n . Требуется вычеркнуть из них минимальное количество чисел так, чтобы оставшиеся шли в порядке возрастания. (Можно решить перебором).

Задача 3. Всем известно, что книга состоит из листов, каждый из которых содержит по две страницы. Все страницы в книге пронумерованы таким образом, что первый лист содержит страницы с номерами 1 и 2, второй лист – с номерами 3 и 4 и так далее до последнего листа. Вам заданы номера двух страниц в книге. Требуется определить количество листов книги, которые располагаются между листами с указанными номерами страниц.

Ограничения Числа не превосходят значения 1018.

Входной файл содержит два натуральных числа A и B – номера страниц в книге.

В выходной файл выведите ответ на задачу.

Пример: входные данные 1 7; выходные данные 2.

2. Поиск в ширину.

Задача 4. В таблице из N строк и N столбцов некоторые клетки заняты шариками, другие свободны. Выбран шарик, который нужно переместить, и место, куда его нужно переместить. Выбранный шарик за один шаг перемещается в соседнюю по горизонтали или вертикали свободную клетку. Требуется выяснить, возможно ли переместить шарик из начальной клетки в заданную, и найти путь из наименьшего количества шагов.

Усложнение: Если некоторые клетки заняты шариками (ходить нельзя).

Решение см. [1] стр. 43, 166-169.

Задача 5. Дана шахматная доска, состоящая из 8×8 клеток. Провести ходом коня путь минимальной длины из одной заданной клетки в другую (заданную).

3. Поиск в глубину.

Задача 6. Лабиринт представляет собой квадрат, состоящий из $N \times N$ сегментов. Каждый из сегментов может быть либо пустым, либо заполненным камнем. Гарантируется, что левый верхний и правый нижний сегменты пусты. Лабиринт обнесен сверху, снизу, слева и справа стенами, оставляющими свободными только левый верхний и правый нижний углы. Директор лабиринта решил покрасить стены лабиринта, видимые изнутри. Помогите ему рассчитать количество краски, необходимой для этого.

Решение см. [1] стр. 43, 169-170.

4. Динамическое программирование

Задача 7. В таблице из N строк и N столбцов клетки заполнены цифрами от 0 до 9. Требуется найти такой путь из клетки $(1,1)$ в клетку (N, N) , чтобы сумма цифр в клетках, через которые он пролегает, была минимальной; из любой клетки ходить можно только вниз или вправо.

Решение см. [1] стр. 30, 112-114.

Задача 8 (задача о куче). Из камней весом P_i ($i = 1..N$) требуется набрать кучу весом ровно W или, если это невозможно максимально, близкую к W (но меньшую, чем W).

Решение см. [2] стр. 124-125.

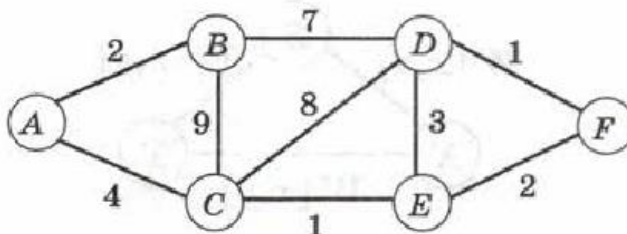
5. Алгоритмы на графах. Минимальное остовное дерево.

Задача 9. В стране Лимония есть N городов, которые нужно соединить линиями связи. Расстояния между городами известно. Между какими городами нужно проложить линии связи, чтобы все города были связаны в одну систему, и общая длина линий связи была наименьшей?

Решение см. [2] стр. 110-112.

6. Алгоритмы на графах. Кратчайший маршрут.

Задача 10. Известна схема дорог между несколькими городами. Числа на схеме (см. рисунок ниже) обозначают расстояния (дороги не прямые, поэтому неравенство треугольника может нарушаться). Нужно найти кратчайший маршрут из города А в город F.



Решение см. [2] стр. 109-110, 112-115.

Литература

1. Меньшиков Ф. В. Олимпиадные задачи по программированию. //СПб.: Питер, 2006.-315с.
 2. Поляков К.Ю., Еремин Е.А. Информатика. Углубленный уровень. Учебник 11 класса: в 2-х частях. Ч2 // М.: Бином. Лаборатория знаний, 2013. – 304с.
-

Задачи Областной олимпиады (II-этап) прошлых лет

Год 2019.

Задача 1. Страницы в журнале (20 баллов)

Текст каждой страницы журнала разделен на три колонки. Колонки пронумерованы следующим образом: первая страница содержит колонки с номерами 1, 2 и 3, вторая страница – колонки с номерами 4, 5 и 6 и так далее. Вам заданы номера двух колонок в журнале. Требуется определить количество страниц журнала, которые располагаются между страницами с указанными номерами колонок (сами страницы с указанными номерами колонок в подсчет не входят).

Внимание! При вводе неверных знаков (например, букв) – предусмотреть обработку исключения, чтобы программа не завершала работу в аварийном режиме.

Формат вывода информации на экран:

Введите два номера колонок – 2

10

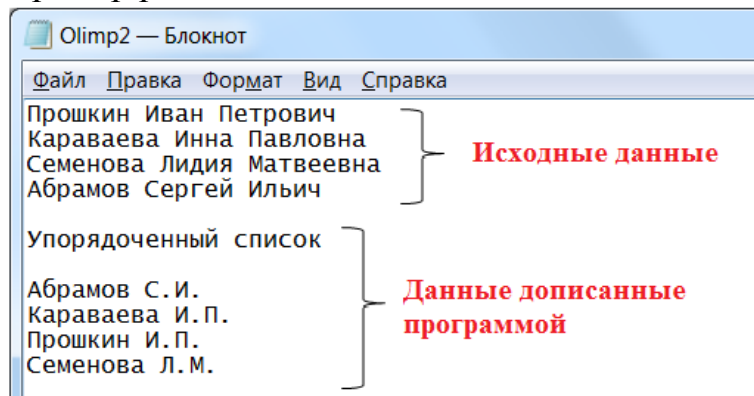
Количество страниц – 2

Задача 2. Упорядоченный список (20 баллов)

В файле input2.txt задан неупорядоченный список участников олимпиады с указанием их фамилий, имен и отчеств (отчество может отсутствовать).

Количество участников в списке считается неизвестным. Требуется составить программу, которая считывает данные из файла input2.txt, преобразует их в формат «Фамилия+Инициалы», упорядочивает полученный список по фамилии «от А до Я» и добавляет полученный список в файл output2.txt. Полученная и записанная информация также выводится на экран.

Пример решения задачи:



Задача 3. Лесной фонд. (30 баллов)

Фотосъемка со спутника представлена в виде матрицы размера $N \times M$ ($N, M < 50$) с элементами из 0 и 1, которая показывает расположение лесных массивов. Лесной массив – это группа соседних клеток из 1, обрамленных со всех сторон 0. Соседними являются клетки, граничащие по горизонтали или вертикали. Количество 1 в группе определяют размер лесного массива. Лесные массивы не соприкасаются друг с другом. Пример такой схемы:

```

00001111
11000011
10000000
00011110
00000000
    
```

Лесные массивы

Требуется написать программу, которая подсчитывает, а затем выводит на экран и в файл output3.txt следующую информацию:

1. Общее количество лесных массивов.
2. Количество 2-х, 3-х и более клеточных лесных массивов.

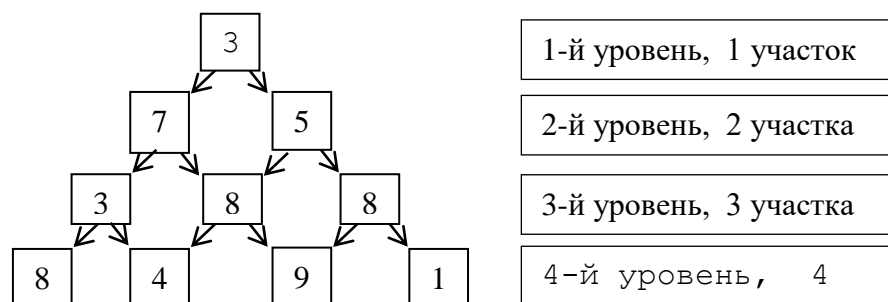
Входные данные. В строках файла input3.txt, через пробел, задаются значения строк матрицы фотосъемки. Количество строк и столбцов – неизвестно.

Выходные данные. Вывести исходную матрицу и требуемую информацию на экран.

Задача 4. Гигантский слалом (30 баллов)

Для проведения соревнований по гигантскому слалому среди мастеров, необходимо спроектировать трассу спуска с горы максимальной сложности. На руках у организаторов есть карта сложности горы, в которой гора разбита на N уровней (отсчет начинается с вершины), каждый уровень разбит на участки с указанием их сложности преодоления. Переход от участка верхнего уровня к

участку нижнего уровня осуществляться вниз влево или вправо по диагонали. Фрагмент такой карты приведен на рисунке:



Требуется написать программу, которая найдет максимально сложный спуск. При этом следует указать общую сложность такого спуска и указать его маршрут (от вершины). Карту горы необходимо скачать из файла INPUT4.TXT.

Входные данные. В файле input4.txt в первой строке записано число уровней N . В последующих N строках, последовательно задаются сложности участков каждого уровня. Например:

INPUT3.TXT
4
3
5 7
3 2 8
8 4 9 1

Выходные данные. Вывести исходные данные и требуемую информацию, которая должна выглядеть так:

Монитор
Общая сложность - 27
Маршрут (от вершины) (1,1) -> (2,1) -> (3,2) -> (4,3)

Здесь (i,j) – обозначение участка, где i – номер уровня, j – номер участка на уровне (отсчет слева на право).

Год 2020.

Задача 1. Несократимая дробь

Пользователь вводит числитель a и знаменатель b дроби. Необходимо привести дробь к несократимому виду. Если $a > b$, то ответ должен содержать целую часть и дробную.

Внимание! При вводе неверных знаков (например, букв) – предусмотреть обработку исключения, чтобы программа не завершала работу в аварийном режиме.

Формат вывода информации на экран:

```
Введите числитель дроби: 8
Введите знаменатель дроби: 28
-----
Несократимая дробь: 2/7
```

```
Введите числитель дроби: 11
Введите знаменатель дроби: 4
-----
Несократимая дробь:
Целая часть: 2. Дробная: 3/4
```

Задача 2. Международный женский день

Михаил на 8 марта решил купить маме, бабушке и девушке по букету из 7 роз. В цветочном магазине есть розы 3-х сортов: чайная, по цене 130,50р., красная – 110,20р., желтая – 88,10р. Два букета считаются разными, если они отличаются по количеству роз какого-то либо сорта.

Требуется:

- 1) Подсчитать количество разных букетов (из 7 роз) и результат вывести на экран.
- 2) Создать список букетов с указанием количества роз разного сорта.
- 3) Рассчитать стоимость каждого букета.
- 4) Вывести на экран список букетов, упорядоченный по убыванию цены.

Формат вывода информации на экран (фрагмент):

```
Всего букетов = 
```

№	Чайных	Красных	Белых	Цена
1	7	0	0	913,50
2	6	1	0	893,20

Задача 3. Словарь ABC

Создаем список всех 5-буквенных слов, составленных из букв {A,B,C}. Например, АСААС, ВАВВС и т.д. Требуется написать программу, которая:

- 1) Подсчитывает количество всех 5-буквенных слов.
- 2) Выводит на экран упорядоченный список всех 5-буквенных слов, в порядке от А до С (по возрастанию).
- 3) Организует поиск слова по номеру в упорядоченном списке. Причем режим поиска продолжается до тех пор, пока не нажата клавиша Escape. При вводе несуществующего номера пишется «Таких слов нет».

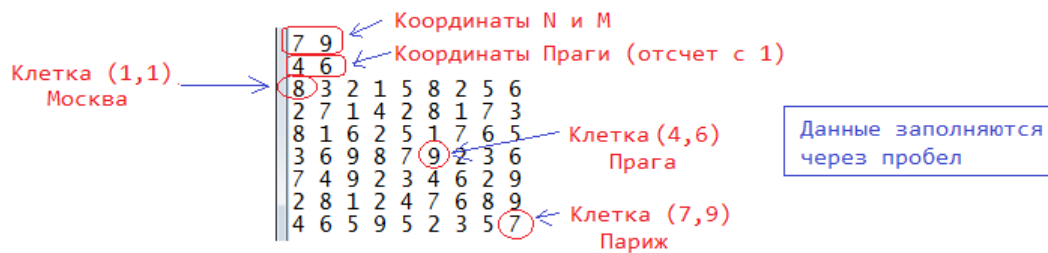
```
Введите номер: 588
Таких слов нет
Выход - Escape, Продолжение - любая
Введите номер: 150
Слово = ВСВВС
Выход - Escape, Продолжение - любая
```

- 4) Организует поиск номера по слову в упорядоченном списке. Режим поиска – пока не нажата клавиша Escape. При вводе несуществующего слова пишется «Таких слов нет».

Задача 4. Путешествие в Париж

Друзья решили совершить путешествие по маршруту Москва-Прага-Париж (посещение Праги – обязательно). Карта выбора маршрута – это таблица, состоящая из N строк и M столбцов. Клетки таблицы заполнены цифрами от 0 до 9 – условные затраты при движении по данному району. Карта находится в файле input4.txt: в первой строке записаны числа N и M, во второй строке – координаты

Праги (отсчет с 1), в остальных N строках – условные затраты. Пример такой карты приведен на рисунке:



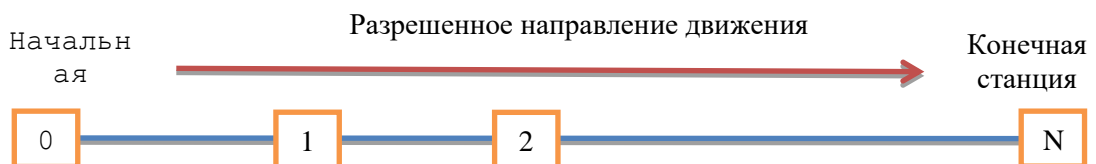
Требуется найти маршрут из Москвы в Париж (через Прагу) с минимальными суммарными затратами. Из любой клетки ходить можно только вниз или вправо.

Формат вывода информации на экран:

- 1) Вывести масштаб карты (числа N и M) и координаты Праги.
- 2) Вывести карту от Москвы до Праги, с учетом возможности проезда до Праги.
- 3) Вывести карту от Праги до Парижа, с учетом возможности проезда до Парижа.
- 4) Вывести суммарные затраты оптимального маршрута.
- 5) Вывести оптимальный маршрут в формате: (1,1)->(1,2)->(2,2)->....->(7,9).

Задача 5. Дешевый путь на дачу

Едим на дачу по прямой ветке электрички от станции 0 к станции N. Электрички ходят со всеми остановками, и есть возможность пересадки. Стоимость проезда между любыми двумя станциями известна. Билеты покупаются на станциях.



Надо найти маршрут, с возможными пересадками, который имеет наименьшую стоимость проезда.

Входные данные – файл input5.txt. Первая строка – число N. Далее, идет стоимость проезда от станции 0 до станций 1,2,3,...,N, затем от станции 1 до станций 2,3,...,N, и последняя строка – стоимость проезда между станциями N-1 и N. Пример файла input5.txt:

```

3 ← Число N - номер конечной станции
|7 10 20 ← Стоимость проезда от станции 0 до станций 1,2,3
|4 8 ← Стоимость проезда от станции 1 до станций 2,3
|2 ← Стоимость проезда от станции 2 до станции 3
    
```

Здесь минимальная стоимость проезда = 12. Оптимальный путь 0->2->3

Выходные данные: Минимальная стоимость проезда. Оптимальный путь.