

## САМОСТОЯТЕЛЬНАЯ РАБОТА №3 (5+5+5 БАЛЛОВ)

Вариант	Задача 1	Задача 2	Задача 3
1	1	CalcTree10	Graf1
2	2	CalcTree9	Graf2
3	3	CalcTree8	Graf3
4	1	CalcTree7	Graf4
5	2	CalcTree6	Graf5
6	3	CalcTree5	Graf6
7	1	CalcTree4	Graf7
8	2	CalcTree3	Graf8
9	3	CalcTree2	Graf9
10	1	CalcTree1	Graf10

### ЗАДАЧА 1

1. Напишите генератор паролей. Составьте три уровня сложности генерации паролей (вместе с их длиной) и спрашивайте у пользователя, какой уровень сложности ему нужен. Проявите свою изобретательность: надёжные пароли должны состоять из сочетания строчных букв, прописных букв, цифр и символов. Пароли должны генерироваться случайным образом каждый раз, когда пользователь запрашивает новый пароль.

2. Ввести с клавиатуры любое слово. Используя **генерацию случайных чисел**, переставить буквы этого слова в случайном порядке. Делать это до тех пор, пока полученное слово не совпадёт с начальным словом. Выводить слово после каждой перестановки и посчитать общее количество выведенных слов (не считая исходного). Пример выполнения программы:

Введите слово: корова

воакро

вокроа

ароовк

краоов

крваоо

оокавр

ооквра

вкраоо

корова

9 попыток

3. Напишите программу-телеграф, которая принимает от пользователя сообщение и выводит его на экран в виде последовательности точек и тире. Азбука Морзе для букв русского алфавита приведена ниже.

Буква	Код	Буква	Код	Буква	Код	Буква	Код
А	·—	Б	—···	В	·— —	Г	— ·
Д	— ·	Е	·	Ж	·· —	З	— · ·
И	··	Й	· — —	К	— · —	Л	·· ·
М	— —	Н	— ·	О	— — —	П	· — — ·
Р	· — ·	С	·· ·	Т	—	У	·· —

Ф	....	Х	....	Ц	....	Ч	....
Ш	....	Щ	....	Ъ	....	Ы	....
Ь	....	Э	....	Ю	....	Я	....

## ЗАДАЧА 2

**CalcTree1.** В текстовом файле с именем filename дано арифметическое выражение в **обратной польской записи**. Операндами являются целые числа из промежутка от 0 до 9. Используемые операции: сложение (+), вычитание (-) и умножение (\*). Постройте дерево, соответствующее данному выражению. Знаки операций кодируйте числами: сложение(-1), вычитание(-2), умножение(-3). Преобразуйте дерево так, чтобы в нем не было операции вычитания (замените поддеревья, в которых есть вычитание значением данного поддерева). Выведите указатель на корень полученного дерева.

**CalcTree2.** В текстовом файле с именем filename дано арифметическое выражение в **обратной польской записи**. Операндами в выражении являются целые числа из промежутка от 0 до 9. Используемые операции: сложение (+), вычитание (-), умножение (\*) и деление нацело (/). Постройте дерево, соответствующее данному выражению. Знаки операций кодируйте числами: сложение(-1), вычитание(-2), умножение(-3), деление нацело (-4). Преобразуйте дерево так, чтобы в нем не было операции сложения (замените поддеревья, в которых есть сложение значением данного поддерева). Выведите указатель на корень полученного дерева.

**CalcTree3.** В текстовом файле с именем filename дано арифметическое выражение в **обратной польской записи**. Операндами в выражении являются целые числа из промежутка от 0 до 9. Используемые операции: сложение(+), вычитание(-), умножение(\*), деление нацело (/) и целочисленный остаток от деления (%). Постройте дерево, соответствующее данному выражению. Знаки операций кодируйте числами: сложение (-1), вычитание (-2), умножение (-3), деление нацело (-4) и целочисленный остаток от деления (-5). Преобразуйте дерево так, чтобы в нем не было операции умножения (замените поддеревья, в которых есть умножение значением данного поддерева). Выведите указатель на корень полученного дерева.

**CalcTree4.** В текстовом файле с именем filename дано арифметическое выражение в **обратной польской записи**. Операндами в выражении являются целые числа из промежутка от 0 до 9. Используемые операции: сложение (+), вычитание (-), умножение (\*), деление нацело (/) и целочисленный остаток от деления (%) и возведение в степень (^). Постройте дерево, соответствующее данному выражению. Знаки операций кодируйте числами: сложение (-1), вычитание (-2), умножение (-3), деление нацело (-4), целочисленный остаток от деления (-5), возведение в степень (-6). Преобразуйте дерево так, чтобы в нем не было операции деления. Иными словами, замените поддеревья, в которых есть операции / или %, значением данного поддерева. Выведите указатель на корень полученного дерева.

**CalcTree5.** В текстовом файле с именем filename дано арифметическое выражение в **префиксной форме**. Операндами в выражении являются целые числа из промежутка от 0 до 9. Используемые операции: сложение (+), вычитание (-), умножение (\*) и деление нацело(/). Постройте дерево, соответствующее данному выражению. Знаки операций кодируйте числами: сложение(-1), вычитание(-2), умножение(-3), деление(-4). Преобразуйте дерево так, чтобы в нем не было операций сложения и вычитания. Иными словами, замените поддеревья, в которых есть сложение или вычитание значением данного поддерева. Выведите указатель на корень полученного дерева.

**CalcTree6.** В текстовом файле с именем filename дано арифметическое выражение в **префиксной форме**. Операндами в выражении являются целые числа из промежутка от 0 до 9. Используемые операции: сложение (+), вычитание (-), умножение (\*), деление нацело (/), целочисленный остаток от деления (%) и возведение в степень (^). Постройте дерево, соответствующее данному

выражению. Знаки операций кодируйте числами: сложение(-1), вычитание(-2), умножение(-3), деление(-4), остаток от деления(-5), возведение в степень (-6). Преобразуйте дерево так, чтобы в нем не было операций возведения в степень (замените поддеревья, в которых есть возведение в степень, значением данного поддерева). Выведите указатель на корень полученного дерева.

**CalcTree7.** В текстовом файле с именем filename дано арифметическое выражение **в префиксной форме**. Операндами в выражении являются целые числа из промежутка от 0 до 9. Используемые операции: сложение(+), вычитание(-), умножение(\*), деление нацело(/), целочисленный остаток от деления(%) и возведение в степень(^). Постройте дерево, соответствующее данному выражению. Знаки операций кодируйте числами: сложение(-1), вычитание(-2), умножение(-3), деление(-4), остаток от деления(-5), возведение в степень (-6). Преобразуйте дерево, вычислив значения всех поддеревьев, для которых результат вычислений является числом из промежутка от 0 до 9 (замените такие поддеревья их значениями). Выведите указатель на корень полученного дерева.

**CalcTree8.** В текстовом файле с именем filename дано арифметическое выражение **в префиксной форме**. Операндами в выражении являются целые числа из промежутка от 0 до 9. Используемые операции: сложение(+), вычитание(-), умножение(\*), деление нацело(/), целочисленный остаток от деления(%) и возведение в степень(^). Постройте дерево, соответствующее данному выражению. Знаки операций кодируйте числами: сложение(-1), вычитание(-2), умножение(-3), деление(-4), остаток от деления(-5), возведение в степень (-6). Преобразуйте дерево, вычислив значения всех поддеревьев, для которых результат вычислений левого или правого поддерева равен нулю (замените такие поддеревья их значениями). Выведите указатель на корень полученного дерева.

**CalcTree9.** В текстовом файле с именем FN1 дано арифметическое выражение **в инфиксной форме**. В выражении могут использоваться операции: сложение(+), вычитание(-), умножение(\*), деление нацело(/), остаток от деления(%), возведение в степень(^), а так же целые числа из промежутка [1; 30] и переменная x. Для операции возведения в степень показатель степени неотрицательное целое число. Постройте дерево выражения. После этого вычислите значение выражения при заданном значении переменной x и выведите результат в текстовый файл с именем FN2. Преобразуйте дерево, заменив все поддеревья вида  $A+x$  на  $x+A$ , где A - некоторое поддерево, а x - переменная. Распечатайте дерево после преобразования в файл FN2 в префиксной и постфиксной форме, а так же в инфиксной форме с избыточными скобками. При наличии нескольких подряд идущих одинаковых операций дерево должно строиться по правилу: операции одинакового приоритета вычисляются по порядку слева направо. Иными словами, выражение  $2+3+4+5$ , например, должно трактоваться как  $((2+3)+4)+5$ , и не может трактоваться как  $(2+3)+(4+5)$  или  $2+(3+(4+5))$ . Результаты всех вычислений, включая промежуточные, принадлежат типу int.

**CalcTree10.** В текстовом файле с именем FN1 дано арифметическое выражение **в инфиксной форме**. В выражении могут использоваться операции: сложение(+), вычитание(-), умножение(\*), деление нацело(/), остаток от деления(%), возведение в степень(^), а так же целые числа из промежутка [1; 30] и переменная x. Для операции возведения в степень показатель степени неотрицательное целое число. Постройте дерево выражения. После этого вычислите значение выражения при заданном значении переменной x и выведите результат в текстовый файл с именем FN2. Преобразуйте дерево, заменив все поддеревья вида  $x*A$  на  $A*x$ , где A - некоторое поддерево, а x - переменная. Распечатайте дерево после преобразования в файл FN2 используя многострочный формат, в котором дерево положено на бок. Каждый уровень дерева выводите в 4-х позициях и используйте выравнивание по правому краю. При наличии нескольких подряд идущих одинаковых операций дерево должно строиться по правилу: операции одинакового приоритета вычисляются по порядку слева направо. Иными словами, выражение  $2+3+4+5$ , например, должно трактоваться как  $((2+3)+4)+5$ , и не может трактоваться как  $(2+3)+(4+5)$  или  $2+(3+(4+5))$ . Результаты всех вычислений, включая промежуточные, принадлежат типу int.

### ЗАДАЧА 3

**Graf1.** Дано описание неориентированного графа в текстовом файле с именем *FileName*. в виде матрицы смежности. Первая строка файла содержит количество вершин графа (n), а следующие n строк содержат матрицу смежности (m),  $m[i][j]=0$ , если ребра между вершинами i и j не существует. Определить степень для каждой вершины графа. Вывести степени вершин, перечисляя их

в порядке возрастания номеров вершин. Если в графе имеются петли, то каждая петля в степени вершины учитывается дважды.

**Graf2.** Дано описание неориентированного графа в текстовом файле с именем *FileName1*. в виде матрицы смежности. Первая строка файла содержит количество вершин графа ( $n$ ), а следующие  $n$  строк содержат матрицу смежности ( $a$ ),  $a[i][j]=0$ , если ребра между вершинами  $i$  и  $j$  не существует. Построить матрицу инцидентности данного графа и вывести ее в файл с именем *FileName2*. Для справки: матрица инцидентности ( $b$ ) имеет размер  $n \times m$ ,  $m$  - число ребер графа,  $b[i][j]=1$ , если ребро  $j$  инцидентно вершине  $i$ , в противном случае  $b[i][j]=0$ . Нумерацию ребер осуществлять в следующем порядке: сначала ребра, инцидентные вершине номер 1, потом ребра инцидентные вершине номер 2 и т.д. до вершины номер  $n$ . Ребра, инцидентные вершине с номером  $i$  перечислять в порядке возрастания номера второй вершины, инцидентной данному ребру. При выводе в первой строке указать размер матрицы инцидентности: числа  $n$  и  $m$ , а в следующих  $n$  строках разместить матрицу инцидентности.

**Graf3.** Дано описание ориентированного графа в текстовом файле с именем *FileName*. в виде матрицы смежности. Первая строка файла содержит количество вершин графа ( $n$ ), а следующие  $n$  строк содержат матрицу смежности ( $m$ ),  $m[i][j]=0$ , если дуги из вершины  $i$  в вершину  $j$  не существует, иначе  $m[i][j]$  хранит вес соответствующей дуги. Выполнить топологическую сортировку графа. В качестве результата вывести номера вершин графа, полученные в результате сортировки. Если на очередном шаге сортировки имелось несколько равноправных вершин перечислять их в порядке убывания номеров вершин. Если топологическую сортировку выполнить невозможно, то вывести "No solution". Сортировку выполнять используя обход в глубину

**Graf4.** Дано описание ориентированного графа в текстовом файле с именем *FileName*. в виде матрицы смежности. Первая строка файла содержит количество вершин графа ( $n$ ), а следующие  $n$  строк содержат матрицу смежности ( $m$ ),  $m[i][j]=0$ , если дуги из вершины  $i$  в вершину  $j$  не существует, иначе  $m[i][j]$  хранит вес соответствующей дуги. Выполнить поиск в ширину от вершины с номером  $k$ . В результате вывести номера вершин графа, достижимые для данной вершины, в порядке их обхода при поиске в ширину. Если на очередном шаге сортировки имелось несколько равноправных вершин, перечислять их в порядке возрастания номеров вершин.

**Graf5.** Юный путешественник решил изучить схему авиационного сообщения. Схема авиационного сообщения задана в текстовом файле с именем *FileName*. в виде матрицы смежности. Первая строка файла содержит количество городов ( $n$ )  $n \leq 25$ , связанных авиационным сообщением, а следующие  $n$  строк хранят матрицу ( $m$ ),  $m[i][j]=0$ , если не имеется возможности перелета из города  $i$  в город  $j$ , иначе  $m[i][j]=1$ . Определить номера городов, в которые из города  $K$  можно долететь менее чем с  $L$  пересадками. Перечислите номера таких городов в порядке возрастания. Нумерация городов начинается с 1. Если таких городов нет, выведите число (-1).

**Graf6.** Юный путешественник решил изучить схему авиационного сообщения. Схема авиационного сообщения задана в текстовом файле с именем *FileName*. в виде матрицы смежности. Первая строка файла содержит количество городов ( $n$ )  $n \leq 25$ , связанных авиационным сообщением, а следующие  $n$  строк хранят матрицу ( $m$ ),  $m[i][j]=0$ , если не имеется возможности перелета из города  $i$  в город  $j$ , иначе  $m[i][j]=1$ . Определить номера городов, в которые из города  $K$  можно долететь ровно с  $L$  пересадками для самого короткого пути. Перечислите номера таких городов в порядке возрастания. Нумерация городов начинается с 1. Если таких городов нет, выведите число (-1).

**Graf7.** Две корпорации хотят разделить сферы влияния, выбрав два разных города для размещения своих штаб-квартир так, чтобы все города, в некоторой округе от штаб-квартиры не были доступны для конкурентов. Схема автомобильного сообщения между городами задана в текстовом файле с именем *FileName* в виде матрицы смежности. Первая строка файла содержит количество городов ( $n$ ,  $n \leq 25$ ), связанных дорогами, а следующие  $n$  строк хранят матрицу ( $m$ ),  $m[i][j]=0$ , если нет дороги из города  $i$  в город  $j$ , иначе  $m[i][j]=1$ . Даны два города-кандидата с номерами  $K1$  и  $K2$  для этих двух штаб-квартир. Определить есть ли города, в которые можно попасть из обоих штаб-квартир, если двигаться от каждой штаб-квартиры не более чем через  $L$  промежуточных городов. Перечислите номера таких городов в порядке возрастания. Нумерация городов начинается с 1. Если таких городов нет, выведите число (-1).

**Graf8.** Юный путешественник решил изучить схему авиационного сообщения. Схема авиационного сообщения задана в текстовом файле с именем *FileName*. в виде матрицы смежности. Первая строка файла содержит количество городов ( $n$ )  $n \leq 15$ , связанных авиационным сообщением, а следующие  $n$  строк хранят матрицу ( $m$ ),  $m[i][j]=0$ , если не имеется возможности перелета из города  $i$  в город  $j$ , иначе  $m[i][j]=1$ . Определить номера городов, в которые из города  $K$  можно долететь не менее чем с  $L$  пересадками и более коротких путей к таким городам не существует. Перечислите номера таких городов в порядке возрастания. Нумерация городов начинается с 1. Если таких городов нет, выведите число (-1).

**Graf9.** Юный путешественник решил изучить схему авиационного сообщения. Схема авиационного сообщения задана в текстовом файле с именем *FileName1*. в виде матрицы смежности. Первая строка файла содержит количество городов ( $n$ )  $n \leq 15$ , связанных авиационным сообщением, а следующие  $n$  строк хранят матрицу ( $m$ ),  $m[i][j]=0$ , если не имеется возможности перелета из города  $i$  в город  $j$ , иначе  $m[i][j]=1$ . Определить сколько есть маршрутов из города  $K1$  в город  $K2$  с  $L$  пересадками. В файл с именем *FileName2* в первой строке выведите число таких маршрутов, а в следующих строках перечислите все такие маршруты в лексикографическом порядке. Маршрут задается перечислением номеров городов, нумерация городов идет с 1. Если таких маршрутов нет, выведите число (-1).

**Graf10.** Юный путешественник решил изучить схему авиационного сообщения. Схема авиационного сообщения задана в текстовом файле с именем *FileName1*. в виде матрицы смежности. Первая строка файла содержит количество городов ( $n$ )  $n \leq 15$ , связанных авиационным сообщением, а следующие  $n$  строк хранят матрицу ( $m$ ),  $m[i][j]=0$ , если не имеется возможности перелета из города  $i$  в город  $j$ , иначе  $m[i][j]=1$ . Определить все маршруты перелета из города  $K1$  в город  $K2$ . В файл с именем *FileName2* в первой строке выведите число таких маршрутов, а в следующих строках перечислите все такие маршруты в порядке от самых коротких к более длинным, маршруты одинаковой длины перечисляйте в лексикографическом порядке. Маршрут задается перечислением номеров городов, нумерация городов идет с 1. Если таких маршрутов нет, выведите число (-1).