

1. Содержательная (исходная) постановка задачи

Предварительные сведения

Рассматриваем матрицу размера $N \times M$ вида:

Матрица в программе представляется в виде двумерного массива (`array[i][k]`).

Например, пусть дана матрица:

В программном языке эта матрица будет представляться в виде массива, элементами которого являются:

```
Array [0][0] = 1; array [0][1] = 2; array [0][2] = 3 ; array [1][0] = 4 ;  
array [1][1] = 5 ; array [1][2] = 6 ; array [2][0] = 7 ; array [2][1] = 8;  
array [2][2] = 9.
```

Ортонормированная (в высшей терминологии ортогональная) матрица - это такая матрица столбцы и строки которой образуют системы ортонормированных векторов, то есть:

где, n -- порядок матрицы, а d_{jk} -- символ Кронекера.

Другими словами, скалярное произведение строки на саму себя равно 1, а на любую другую строку -- 0. Так же и для столбцов.

Постановка задачи

Матрица является ортонормированной, если скалярное произведение каждой пары различных строк равно 0, а скалярное произведение каждой строки на себя равно 1. Определить, является ли заданная матрица A размера $N \times M$ ортонормированной.

Указания

Использовать две функции:

1. Функция которая проверяет условие ортонормированности матрицы и возвращает `true`, либо `false`
2. Функция, которая непосредственно производит перемножение строк.

2. Анализ и пример решения задачи

Анализ задачи

Как упоминалось ранее для определения ортонормированности функции нужно проверить скалярное произведение строки саму на себя и затем произведение этой строки с последующими.

- Чтобы в программе не выполнялись лишние действие перемножения строки с другими надо начинать не с первой строки а с последующий (соответственно, когда мы в цикле дойдем до последней строки, то она только перемножается сама на себя).
- Перемножение строк реализуется в функции с помощью сложения всех пар произведений типа `array[i][k]*array[j][k]` и записи получившийся суммы в отдельную переменную.
- В проверяющей функции после каждого полученного результата перемножения строк сверяем полученный результат с условием, если неверно, то возвращаем `false`. А после окончания цикла проверки возвращаем `true` (таким образом, если матрица ортонормирована, то она пройдет все проверки и вернет `true`)

Пример решения задачи

Рассмотрим 2 примера решения задачи, для матрица, который является и не

является ортонормированной.

1. Матрица ортонормированна. Сначала находим произведение первой строки саму на себя:

$$S = 1*1 + 0*0 + 0*0 + 0*0 = 1 \text{ (удовлетворяет условию)}$$

· Затем происходит перемножение первой строки на 2,3 и 4

$$S_{1,2} = 0; S_{1,3} = 0; S_{1,4} = 0 \text{ (удовлетворяет условию)}$$

· Далее рассматриваем вторую строку (во-первых производим умножение саму на себя, а затем рассматриваем произведения уже со строками 3 и 4)

$$S_{2,2} = 1; S_{2,3} = 0; S_{2,4} = 0 \text{ (удовлетворяет условию)}$$

· По аналогии рассматриваем третью строку.

$$S_{3,2} = 1; S_{3,4} = 0 \text{ (удовлетворяет условию)}$$

· Для последней строки производи только перемножение саму на себя

$$S_{4,2} = 1 \text{ (удовлетворяет условию)}$$

· Матрица прошла все проверки, значит она ортонормирована.

2. Матрица не ортонормирована. Сначала находим произведение первой строки саму на себя:

$$S = 1*1 + 0*0 + 0*0 + 0*0 = 1 \text{ (удовлетворяет условию)}$$

· Затем происходит перемножение первой строки на 2,3 и 4

$$S_{1,2} = 1*0 + 0*2 + 0*0 + 0*1 = 0; \text{ (удовлетворяет условию)}$$

$$S_{1,3} = 1*3 + 0*5 + 0*3 + 0*6 = 3 \text{ (не удовлетворяет условию)}$$

· Функция возвращает false значит матрица не ортонормирована.

3. Формальная постановка задачи

Исходные данные

1) Размеры матрицы целые числа n и m .

2) Двумерный массив размера $n \times m$ содержащий элементы вещественные числа двойной точности.

Ограничения на исходные данные

Числа n и m должны быть меньше 50, т.к. агау это статический массив размера 50x50.

Матрица состоит из вещественных чисел

Результирующие (выходные) данные

Выходными данными является отчет о том ортонормированная ли матрица.

Связь выходных данных с исходными данными

Опишем формально связь между выходным булевским значением `ответ` и входной матрицей `a`. Для этого используем следующие функции над матрицей.

1) Функция определяющая скалярное произведение строки на строку в математическом представлении выглядит следующим образом:

где m - это количество столбцов в матрице

2) Функция проверки ортонормированности матрицы:

Перебор строк от $i = 1$ до n

$j = i$;

Если $S \neq 1$ (значение S берется из первой функции)

то Функция = false;

Перебор строк от $j = i + 1$ до $n - 1$

Если $S \neq 0$

то Функция = false;

Конец перебора;

Конец перебора;

Функция = true;

Т.е. входная матрица проверяется в функции 2, для реализации скалярного произведения строк используется функция 1. В проверяющей функции рассматриваются все возможные варианты скалярного произведения строк и сравниваются с условием задания. Если хоть одно условие не совпадает, то функция завершается и принимает false. Если матрица успешно проходит все проверки, то по завершению начального цикла функции присваивается значение true.

4. Спецификация программы

Исходные данные (ИД)

Перечень и основные характеристики ИД:

- 1) n - количество строк в матрице
- 2) m - количество столбцов в матрице
- 3) Array - Массив, задающий проверяемую матрицу A , вида

Ограничения на исходные данные:

- 1) Для n и m задан диапазон $2 \leq n, m \leq 50$, это натуральные числа типа int
- 2) Элементы вводимого массива любые числа диапазона double

Место и форма представления исходных данных:

- 1) n и m вводится с клавиатуры в процессе диалога, порядок ввода и форма представления описаны в сценарии диалога;
 - 2) данные, задающие матрицу, вводятся с клавиатуры в процессе диалога через пробел с переносом на следующую строку, тогда когда все элементы строки будут введены. Введённая матрица будет выглядеть следующим образом
- Функции программы по обработке исключительных ситуаций

Исполняющая система прекращает выполнение программы, выдавая

диагностическое сообщение об ошибке в следующих случаях:

- 1) если значение n не принадлежит диапазону $2 \dots 50$;
- 2) если значение m не принадлежит диапазону $2 \dots 50$;

Выходные данные

Состав выходных данных:

- 1) Сообщения, которые просят пользователя ввести размеры матрицы.
- 2) Если размер массива не вписывается в рамки 50 на 50 , то выводятся сообщения об ошибке.
- 3) Сообщение, которое просит пользователя последовательно ввести элементы матрицы.
- 4) Отчет о проверке матрицы.

Место и форма представления выходных данных. Выходные данные выводятся на экран.

- 1) Вывод сообщений диалога пользователя по порядку подробнее описано в пункте

описание сцен диалога.

Макет вывода выходных данных в диалоговом окне в том случае, когда матрица ортонормированна
Сценарий диалога

Общая схема диалога:

- 1.Сцена 1 (ввод, анализ количества строк в матрице + если размер недопустим выводится сообщение об ошибке).
- 2.Сцена 2 (ввод, анализ количества столбцов в матрице + если размер недопустим выводится сообщение об ошибке).
- 3.Сцена 3 (ввод элементов матрицы).
- 4.Сцена 4 (вывод отчета).

Описание сцен диалога. Описание сцены 1 приведено в табл. П.1.

Таблица П.1

Алгоритм ведения диалога

Сообщения пользователю и его реакция

Вывод запроса

Enter number of rows

Ввод n

??<Enter>

Если размер неверный

Matrix size is inappropriate

Описание сцены 2 аналогично сцене 1 приведено в табл. П.2.
Таблица П.2

Алгоритм ведения диалога

Сообщения пользователю и его реакция

Вывод запроса

Enter Number of Columns

Ввод m

??<Enter>

Если размер неверный

Matrix size is inappropriate

Описание сцены 3 приведено в табл. П.3.
Таблица П.3

Алгоритм ведения диалога

Сообщения пользователю и его реакция

Вывод запроса

Lead elements of the matrix

Ввод массива array

??...<Space>...??...<Space>

??...<Space>...??...<Space>

.

..

??...<Space>...??...<Space>

5. Разработка структур данных и алгоритмов

Алгоритм тела программы

Алгоритм функции summa:

6. Код программы на языке C++

```
#include <iostream>
using namespace std;
/*
* Функция выполняет скалярное перемножение двух строк матрицы
* m - количество столбцов в матрице
* i - номер первой перемножаемой строки
* j - номер второй перемножаемой строки
* k - интератор столбца
* array - проверяемый массив
* s - сумма произведений пар элементов
* Функция возвращает значение переменной s
*/
double summa(double array[50][50], int m, int i, int j);
/*
* Функция выполняет проверку ортонормированности матрицы
* n - количество строк в матрице
* i - интератор основной строки
* j - интератор второй строки
* array - проверяемый массив
* s - значение, которое возвращает функция summa,
* обозначающее результат перемножения двух строк
* Функция возвращает true, если матрица ортонормированна
* и false в противоположном случае
*/
bool proverka (double array[50][50], int n, int m);
int main()
{
double array[50][50]={0};
int n=0,m=0;//максимальное количество строк и столбцов
bool otvet=0;//булевская переменная, отображающая результат проверки массива
cout << "Enter number of rows" << endl;
cin >> n;// Вводим количество строк
/*если количество строк больше, чем максимальное количество элементов,
выводим сообщение о превышении максимального значения и завершаем работу
программы */
if (n>50||n<2){
cout << "Matrix size is inappropriate" << endl;
```

```

return 0;}
cout << "Enter Number of Columns" << endl;
cin >> m;// Вводим количество столбцов
/*если количество столбцов больше, чем максимальное количество элементов,
выводим сообщение о превышении максимального значения и завершаем работу
программы */
if (m>50||m<2){
cout << "Matrix size is inappropriate" << endl;
return 0;}
cout << "Lead elements of the matrix" << endl;
//Осуществляем ввод элементов массива
for(int i=0;i<n;i++)
{
for(int k=0;k<m;k++)
cin >> array[i][k];
cout << endl;
}
otvet=proverka (array, n, m);//Вызывает функцию проверки и возвращаемое значение
присваиваем булевской переменной otvet
//Выводим сообщение о результате проверки
if (otvet==true)
cout << "Matrix is orthonormal";//Выводим сообщение об ортонормированности
матрицы
else cout << "Matrix is not orthonormal";//Выводим сообщение о том, что матрица не
ортонормированна
getch();
return 0;
}
bool proverka (double array[50][50], int n, int m)
{
int i=0, j=0;
double s=0;
// начало проверяющего цикла где сначала проверяем строки с 0 до n-1
for (i=0; i<n; i++){
// передаем дополнительной строке значение основной, для проверки скалярного
произведения строки саму на себя
j=i;
s=summa(array, m, i, j);
// проверка условия
if (s!=1)
// если сумма не равна 1 то возвращаем false
return false;
// перебираем дополнительные строки начиная с i + 1

```



```
for (j=i+1; j<n; j++){
s=summa(array, m, i, j);
// проверка условия
if (s!=0)
// если сумма не равна 0 то возвращаем false
return false;}}
// после окончания проверки возвратим true
return true;
}
double summa(double array[50][50], int m, int i, int j)
{
double s=0;
for (int k=0; k<m; k++)
// суммируем произведение пар элементов
s+=array[i][k]*array[j][k];
// в конечном итоге возвращаем результат скалярного произведения двух строк
return s;
}
7. Испытание программы
```

n = 54

Matrix size is inappropriate

n = 12

m = -3

Matrix size is inappropriate

n = 4

```
m = 5
array =
1 0 3 7 8
9 0 0 5 1
3 8 0 6 1
0 0 0 0 1
```

Matrix is not orthonormal

```
n = 4
m = 5
array =
1 0 0 0 0
0 1 0 0 0
0 0 0 0 1
0 0 1 0 0
```

Matrix is orthonormal

```
n = 2
m = 2
array =
1 0
0 1
```

Matrix is orthonormal

ортонормированная матрица программа

8. Анализ результатов

Программа прошла тестирование со всеми возможными входными данными, а так же проверкой всех диалогов между пользователем-программой. В процессе тестирования некорректной работы программы не выявлено. Следовательно программа может быть признана завершенной и годной к эксплуатации.