

**Открытый урок****«Двумерные массивы»****Цели:**

**образовательные:** способствовать формированию представления о двумерном массиве и основных свойствах квадратных матриц; рассмотреть действия, которые можно выполнить над матрицами; познакомить с типовыми алгоритмами обработки матриц на языке Паскаль; создать условия для формирования умения решать алгоритмы с двумерными массивами;

**развивающие:** создать условия для развития умений работать за ограниченный промежуток времени, развития навыков самостоятельной работы, развития навыков работы с конспектом.

**воспитательные:** содействовать созданию на уроке психологического комфорта для работы, воспитанию собранности, ответственного отношения к своему труду, повышения мотивации к учебной деятельности.

**Тип урока:** комбинированный.

**Методы обучения:** лекция, самостоятельная работа.

**Формы организации:** фронтальная работа, индивидуальная работа.

**Оборудование:** задания для работы в классе (инструкции к выполнению задания в электронном виде, среда программирования Borland Pascal), интерактивная презентация.

**Ход урока:**

1. Организационный момент: Приветствие. Сообщается тема. Ставится цель урока и задачи.

2. Актуализация полученных знаний. Устно проговариваются основные моменты предыдущей темы: «Одномерные массивы». Фронтальный мини опрос по основным определениям.

3. Целеполагание, постановка задач:

- ✓ дать определение понятию двумерный массив;
- ✓ рассмотреть форму записи двумерного массива
- ✓ рассмотреть основные свойства квадратных матриц;
- ✓ рассмотреть действия, которые можно выполнить над матрицами;
- ✓ познакомиться с типовыми алгоритмами обработки матриц на языке Pascal;
- ✓ создать условия для формирования умения решать алгоритмы с двумерными массивами;

**4. Лекция**

**Определение:** Двумерным массивом называется совокупность данных, каждое значение которых зависит от двух чисел – индексов строки и столбца.

**Формат записи**

<i>Способ 1</i>	<i>Способ 2</i>
Var A: array[1..4,1..3] of integer;	Const n=4; m=3; Type mas = Array[1..n, 1..m] of Integer; Var a: mas;

Для того чтобы использовать элемент массива, надо указать имя массива и индексы элемента. **Первый индекс соответствует номеру строки, второй – номеру столбца.**

Например:

```
for i:=1 to n do
  for j:=1 to m do
    a[i,j]:= random (100);
```

Матрица, у которой число строк равно числу столбцов, называется **квадратной**.

$A(n,n)$  – квадратная матрица.

**Основные свойства квадратных матриц:**

1. Квадратные матрицы имеют **главную** и **побочные** диагонали. Например, для матрицы  $A$  на главной диагонали лежат элементы 1,5,9, а на побочной – 3, 5, 7.

$$A = \begin{pmatrix} 123 \\ 456 \\ 789 \end{pmatrix}$$

Если:

- $i=j$  – элементы расположены на главной диагонали;
- $i>j$  – элементы расположены ниже главной диагонали;
- $i<j$  – элементы расположены выше главной диагонали;
- $i \geq j$  – элементы расположены на главной диагонали и ниже;
- $i+j=n+1$  – элементы расположены на побочной диагонали;
- $i+j < n+1$  – элементы расположены над побочной диагональю;
- $i+j > n+1$  – элементы расположены под побочной диагональю;

2. Квадратная матрица, у которой все элементы, исключая элементы главной диагонали, равны нулю, называется **диагональной** матрицей

$$A = \begin{pmatrix} 100 \\ 050 \\ 009 \end{pmatrix}$$

3. Диагональная матрица, у которой все элементы, стоящие на главной диагонали, равны 1, называется **единичной** матрицей

$$A = \begin{pmatrix} 100 \\ 010 \\ 001 \end{pmatrix}$$

4. Если в матрице поменять местами строки и столбцы, то получится матрица, которая называется **транспонированной** матрицей.

### Основные действия, которые можно выполнять над матрицами:

- ✓ суммировать;
- ✓ находить разность;
- ✓ произведение матрицы на некоторое число;
- ✓ произведение двух матриц.

### Типовые алгоритмы обработки матриц на языке Паскаль

#### Вывод матрицы в виде таблицы:

1-й способ

```
for i:= 1 to n do
begin
for j:= 1 to m do write (a[i,j]:4);
writeln
end;
```

2-ой способ

```
for i:= 1 to n do
for j:= 1 to m do
if j>m then write (a[i,j]:4)
else writeln (a[i,j]:4);
```

#### Использование генератора случайных чисел:

```
randomize;
for i:=1 to m do
begin
for j:=1 to n do
begin
a[i,j]:=random(100)-10;
write(a[i,j]:4);
end;
writeln;
end;
```

**Random(x)** - случайное число от 0 до x-1

**Random(101)** – случайное число от 0 до 100

**Random(11)-20** – случайное число от -10 до 10

#### Суммирование матриц:

```
for i:= 1 to n do
begin
for j:= 1 to m do
c[i,j]:=a[i,j]+ b[i,j]
end;
```

**Транспонирование матрицы** – зеркальное отражение её элементов, относительно главной диагонали. Сделать это можно, введя новый массив:

```
for i:= 1 to n do
for j:= 1 to n do
b[i,j]= a[j,i];
```

**5. Первичный контроль****1 вариант**

1. Исправьте неточности в описании двумерного массива:

```
Var
  A= array[1..n,1..m] of integer;
```

2. Вставьте пропущенные определения:

Основные действия, которые можно выполнять над матрицами: суммирование, произведение двух матриц, \_\_\_\_\_, \_\_\_\_\_.

3. Вставьте пропущенные определения:

Матрица, у которой число строк равно числу столбцов, называется \_\_\_\_\_.

4. Найдите ошибки в алгоритме:

```
for i:=1 to n do
begin
  for j:=1 to m do
    c[i,j]:=a[i,j]+ a[i,j]
  end;
end;
```

Какую типовую задачу выполняет алгоритм?

**2 вариант**

1. Исправьте неточности в описании массива:

```
Const n=4; m=3;
type
  mas=array[1..m,1..n] of integer;
```

2. Вставьте пропущенные определения:

Диагональная матрица, у которой все элементы, стоящие на главной диагонали, равны \_\_\_\_\_, называется единичной матрицей

3. Вставьте пропущенные определения:

Квадратная матрица, у которой все элементы, исключая элементы главной диагонали, равны нулю, называется \_\_\_\_\_.

4. Найдите ошибки в алгоритме:

```
for i:=1 to m do begin
  for j:=1 to n do begin
    a[i,j]:=random(100)-10;
  end; end;
```

Какую типовую задачу выполняет алгоритм?

**Ответы:****1 вариант**

1. `Var A:array[1..n,1..m] of integer;`
2. Основные действия, которые можно выполнять над матрицами: суммирование, произведение двух матриц, **нахождение разности двух матриц, произведение матрицы на некоторое число.**
3. Матрица, у которой число строк равно числу столбцов, называется **квадратной** матрицей.
4. Типовая задача: **Суммирование двух матриц.**  

```
for i:= 1 to n do
begin
  for j:= 1 to m do
    c[i,j]:=a[i,j]+ b[i,j]
  end;
```

**2 вариант**

1. `const n=4; m=3;`  
`type mas=array[1..m,1..n] of integer;`
2. Диагональная матрица, у которой все элементы, стоящие на главной диагонали, равны 1, называется **единичной** матрицей
3. Квадратная матрица, у которой все элементы, исключая элементы главной диагонали, равны нулю, называется **диагональной** матрицей .
4. Типовая задача: **Использование генератора случайных чисел**  

```
for i:=1 to m do begin
  for j:=1 to n do begin
    a[i,j]:=random(100)-10;
    write(a[i,j]:4);
  end;writeln; end;
```

**6. Вторичный контроль****Задачи с матрицами\*\*\***

1. Написать программу, которая для целочисленной матрицы 3x4 определяет среднее арифметическое ее элементов и количество положительных элементов в каждой строке.

```
program sred_n;
const m=3; n= 4;
var a: array[1..m,1..n] of integer;
    i,j,k_pol_el:integer; sred:real;
```

```

begin
for i:=1 to m do
for j:=1 to n do read(a[i,j]);
sred:=0;
for i:=1 to m do begin
k_pol_el:=0;
for j:=1 to n do begin
sred:=sred+a[i,j];
if a[i,j]>0 then inc(k_pol_el);
end;
writeln('В',i,'-ой строке ',k_pol_el,' положительных элементов');
end;
sred:=sred/m/n;
writeln('Среднее арифметическое всех элементов матрицы:',sred:6:2);
readLn
end.

```

2. Написать программу, которая для прямоугольной целочисленной матрицы 3x4 определяет номер самого левого столбца, содержащего только положительные элементы. Если такого столбца нет, выводится сообщение.

```

program num_posit;
const m=3; n=4;
var a: array[1..m, 1..n] of integer;
    i,j,num:integer; all_posit:boolean;
begin randomize;
for i:=1 to m do begin
for j:=1 to n do begin
a[i,j]:=random(101)-200;
write(a[i,j]:4);
end; writeln; end;
    num:=0;
    for j:=1 to n do
    begin
        all_posit:=true;
        for i:=1 to m do
            if a[i,j] < 0 then begin all_posit:=false; break; end;
        if all_posit then begin num:=j; break; end;
    end;
if num = 0 then writeln('Положительных столбцов нет')
else writeln('Номер положительного столбца ',num);
ReadLn; end.

```

\*\*\*Учащиеся на своих рабочих местах загружают среду программирования Паскаль. Учитель вызывает к головному компьютеру ученика. На интерактивной доске будут отражаться все его действия, а учитель по ходу создания программы будет «наводить» на правильные действия при выборе алгоритма. Ученики просматривают результаты выполнения программы, анализируют результаты, вносят коррективы, отвечают на дополнительные вопросы:

1. Как организуется двумерный массив?
2. Что означает процедура inc?
3. Что означает процедура break?
4. Как используется генератор случайных чисел?

## **7. Подведение итогов. Оценка уровня усвоения.**

**ДЗ:** конспект, 2-3 задачи: по индивидуальным карточкам:

3. Дана матрица  $A_{5 \times 5}$ , содержащая случайные элементы. Найти сумму всех элементов матрицы.

4. Найти сумму положительных элементов указанного столбца матрицы  $A_{5 \times 5}$  целых чисел.
5. Дан двумерный квадратный массив. Найти номера строк, все элементы которых равны нулю.
6. Дан двумерный квадратный массив. Найти номера строк, элементы в каждой из которых одинаковы между собой.
7. Определить минимальный элемент двумерного массива. Напечатать номер строки, содержащей максимальное число минимальных элементов, если такие имеются.
8. Дан двумерный массив. Найдите строку с наибольшей суммой элементов и наименьшей. Вывести на экран найденные строки и суммы их элементов.
9. Вычислить сумму элементов матрицы по столбцам.
10. Написать программу, которая вычисляет сумму диагональных элементов квадратной матрицы.
11. Найти произведение ненулевых элементов матрицы.