Практическая работа 4.1. Работа с двумеными массивами.

1)теоретическая часть

Рассмотрим программный код для работы с одномерным массивом:

// Объявление одномерного массива

// чисел целого типа

int[] mas;

// Создание одномерного массива из

// десяти элементов чисел целого типа

mas = new int[10];

// Запись значения в элемент

// одномерного массива

mas[0] = 10;

mas[1] = 20;

mas[9] = 90;

Одномерный массив целых чисел напоминает ряд шкафчиков, в каждый из которых можно поместить некоторое целое число. Каждый из шкафчиков имеет свой номер в виде целого положительного числа. Нумерация элементов массива начинается с нуля! Именно по этому номеру можно обратиться к нужному элементу массива. Номер элемента массива указывается в квадратных скобках. Можно сказать, что массив –это группа переменных одинакового типа. В данном примере представлен массив целых чисел типа: int. Для обработки данных массивов часто используются циклы, в частности,известный нам цикл for:

// Запись одинакового значения во все элементы массива

for (int i=0;i<10;i++)

{

mas[i] = 3;

}

Кроме одномерных массивов в программировании также используются двумерные массивы. Двумерный массив представляет из себя таблицу, в которой имеются строки и столбцы. Можно сказать,что двумерный массив –это несколько одномерных массивов, которые расположены в несколько строк (см. рис.2).



Рис. 2

В двумерном массиве положение элемента определяется номером строки и номером колонки (номером элемента в строке).Обратите внимание,что нумерация строк и колонок начинается с нуля! Программный код для работы с двумерным массивом выглядит так:

// Объявление двумерного массива

// чисел целого типа

int[][] mas;

// Создание двумерного массива из трех строк

// и десяти элементов чисел целого типа в каждой строке

mas = new int[3][10];

// Запись значения в элемент

// двумерного массива

mas[0][1] = 10;

mas[1][3] = 20;

mas[2][9] = 90;

Обратите внимание: для задания элемента двумерного массива используется пара квадратных скобок,стоящих рядом. В первых квадратных скобках указывается номер строки, во вторых квадратных скобках номер колонки (номер элемента в строке) (см. рис. 3).

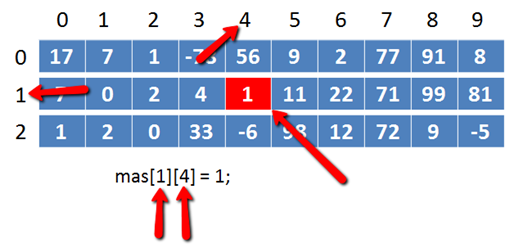


Рис. 3

Для обработки значений элементов двумерного массива используются циклы. В отличие от одномерного массива, в двумерном массиве нужно перебирать в цикле строки и элементы в каждой строке. Поэтому такая задача решается вложенными циклами:

// Внешний цикл перебирает строки массива

for (int i=0;i<3;i++)

{

// Внутренний цикл перебирает элементы в строке массива

for (int j=0;j<10;j++)

{

mas[i][j] = 3;

}

}

Этот фрагмент программного кода перебирает все элементы двумерного массива (см. рис. 2)и вписывает во все элементы одинаковое значение - 3(см.рис. 4).

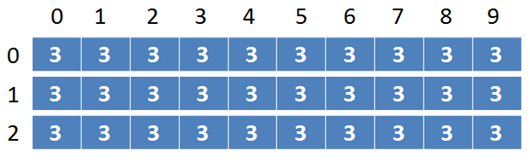


Рис. 4

Рассмотрим более подробно работу этих циклов. Внешний цикл(верхний) будет перебирать строки, а внутренний цикл (нижний) будет перебирать элементы в каждой строке. Обратите внимание на то, что имя счетчика внешнего цикла- i, а имя счетчика внутреннего цикла- j. Это два разных имени переменных. Переменная i отвечает за изменение номера строки, а переменная jотвечает за номер элемента в строке (номер столбца). Работа этих двух циклов по перебору элементов двумерного массива будет выглядеть как чтение книги: элементы будут перебираться построчно сверху вниз, а в каждой строке слева направо (см. рис.5).

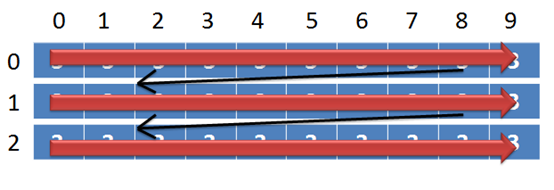


Рис. 5

Давайте пройдем пошагово работу этих двух вложенных циклов.

Первый шаг: i=0; j=0; это самая верхняя строка и самый левый элемент в строке.

Второй шаг: i=0; j=1; это самая верхняя строка и второй слева элемент в строке.

Третий шаг: i=0; j=2; это самая верхняя строка и третий слева элемент в строке.

Два вложенных цикла работают по такой схеме: на единицу увеличивается счетчик iвнешнего (верхнего)цикла, после этого внутренний (нижний) цикл изменяет значение своего счетчика j от начала до конца.Внешний цикл ждет, пока внутренний не закончит свою работу. После этого внешний цикл увеличивает значение своего счетчикаi еще на единицу и снова передаётуправление внутреннему циклу,который изменяет значение своего счётчикаjот начала до конца (с 0 до 9). Получается, что счетчикi внешнего цикла пробежит от начала до конца всего один раз,а счетчик jвнутреннего цикла пробежит от начала до конца много раз!

Теперь запишем и выполним следующую программу работы с двумерным массивом:

//Объявление двумерного массива

// из трех строк и пяти элементов

// в каждой строке

int[][] mas = {{1,1,2,2,2},

{3,3,3,4,4},

{5,5,6,6,6}};

// Внешний цикл перебирает строки массива

for (int i=0;i<3;i++)

{

// Внутренний цикл перебирает элементы в каждой строке

for (int j=0;j<5;j++)

{

// Выводим значение в консольное окно

System.out.print(mas[i][j]);

// Выводим пробел для отступа между числами

System.out.print(" ");

}

//Делаем перевод на новую строку

System.out.println();

}

Рассмотрим программный код подробнее.

При объявлении двумерный массив можно сразу заполнить значениями:

int[][] mas = {{1,1,2,2,2},

{3,3,3,4,4},

{5,5,6,6,6}};

В этом случае не нужно создавать массив при помощи new:

mas = new int[3][5]; // В данном случае не нужно!

Массив с конкретными значениями задается при помощи фигурных скобок. Внутри фигурных скобок задаются строки массива. Каждая строка, в свою очередь, располагается в своих фигурных скобках. Значения элементов каждой строки перечисляются через запятую. Строки друг от друга также разделяются запятыми. При явном указании значений –размер массива определяется автоматически.

После объявления массива с явным указанием значений –выводим его в консольное окно. Для перебора всех значений массива используем два вложенных цикла:

// Внешний цикл перебирает строки массива

for (int i=0;i<3;i++)

{

// Внутренний цикл перебирает элементы в каждой строке

for (int j=0;j<5;j++)

{

// Выводим значение в консольное окно

System.out.print(mas[i][j]);

// Выводим пробел для отступа между числами

System.out.print(" ");

}

//Делаем перевод на новую строку

System.out.println();

}

Метод print() выводит значение в консольное окно, метод println()делает переход на следующую строку. Обратите внимание, что переход на следующую строку выполняется после внутреннего цикла. В тот момент, когда внутренний цикл закончил вывод очередной строки.

Также как и одномерные массивы, двумерные массивы могут быть различных типов данных.

Двумерный массив мы рассматривали не просто так. При разработке игр использование двумерного массива очень часто бывает удобным. Почему?На самом деле игровое поле, при разработке двухмерной компьютерной игры,представляет собой плоскость. Таблица,которую напоминает двумерный массив, также представлена плоскостью. При работе с плоскостью мы имеем дело с двумя координатами - x,y.Расположение каждой точки игрового поля также характеризуется всего двумя значениями - x,y. Двумя значениями определяется и расположение элемента в двумерном массиве. Именно поэтому информацию об игровом поле удобно хранить в двумерном массиве.

Допустим, создается двухмерная игра с лабиринтом в десять блоков по ширине и десять по высоте. На игровом поле имеются препятствия (см. рис. 6).

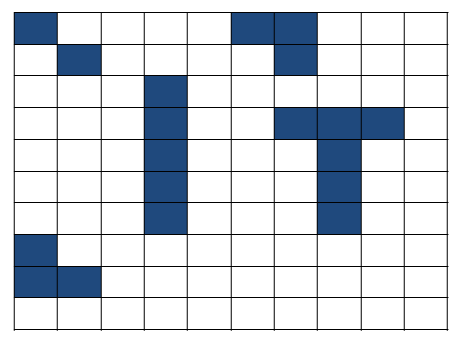


Рис. 6

Создадим модель данного игрового поля при помощи двумерного массива. Для этого нам понадобится массив из десяти строк и десяти элементов в каждой строке (10x10). Для этих целей достаточно обычного двумерного массива целых чисел,типа int. Пусть свободные ячейки игрового поля будут помечаться нулевым значением, а ячейки с преградами будут помечаться единицами.Тогда массив будет выглядеть так:

//Двумерный массив целых чисел 10x10

// хранит данные игрового поля

int[][] mas = {

{1,0,0,0,0,1,1,0,0,0},

{0,1,0,0,0,0,1,0,0,0},

{0,0,0,1,0,0,0,0,0,0},

{0,0,0,1,0,0,1,1,1,0},

{0,0,0,1,0,0,0,1,0,0},

{0,0,0,1,0,0,0,1,0,0},

{0,0,0,1,0,0,0,1,0,0},

{1,0,0,0,0,0,0,0,0,0},

{1,1,0,0,0,0,0,0,0,0},

{0,0,0,0,0,0,0,0,0,0}

};

Если тщательно посмотреть на значения элементов массива,то можно заметить, что расположение единиц в массиве соответствует расположению преград (см. рис. 6). Теперь можно выполнить отрисовку игрового поля по значениям элементов этого массива. Информацию о состоянии игрового поля при создании в двумерном массиве.

2) практическая часть

2.1. Запустите следующий код создания двумерного массива, заполненного нулями:

**int** [][]mas = **new** **int** [24][24];

**for** (**int** i=0; i<24; i++) {

**for** (**int** j=0; j<24; j++) {

mas[i][j]=0;

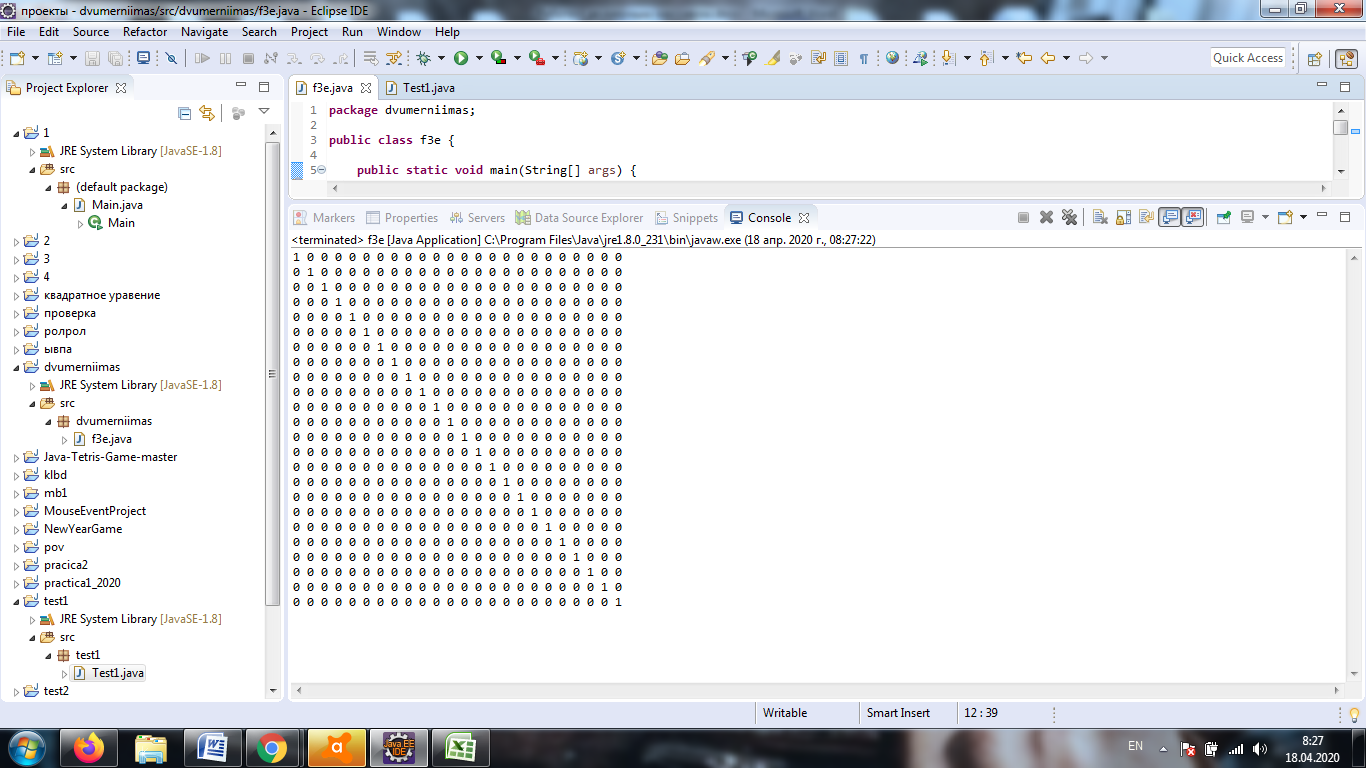
System.***out***.print(mas[i][j]+" ");

}

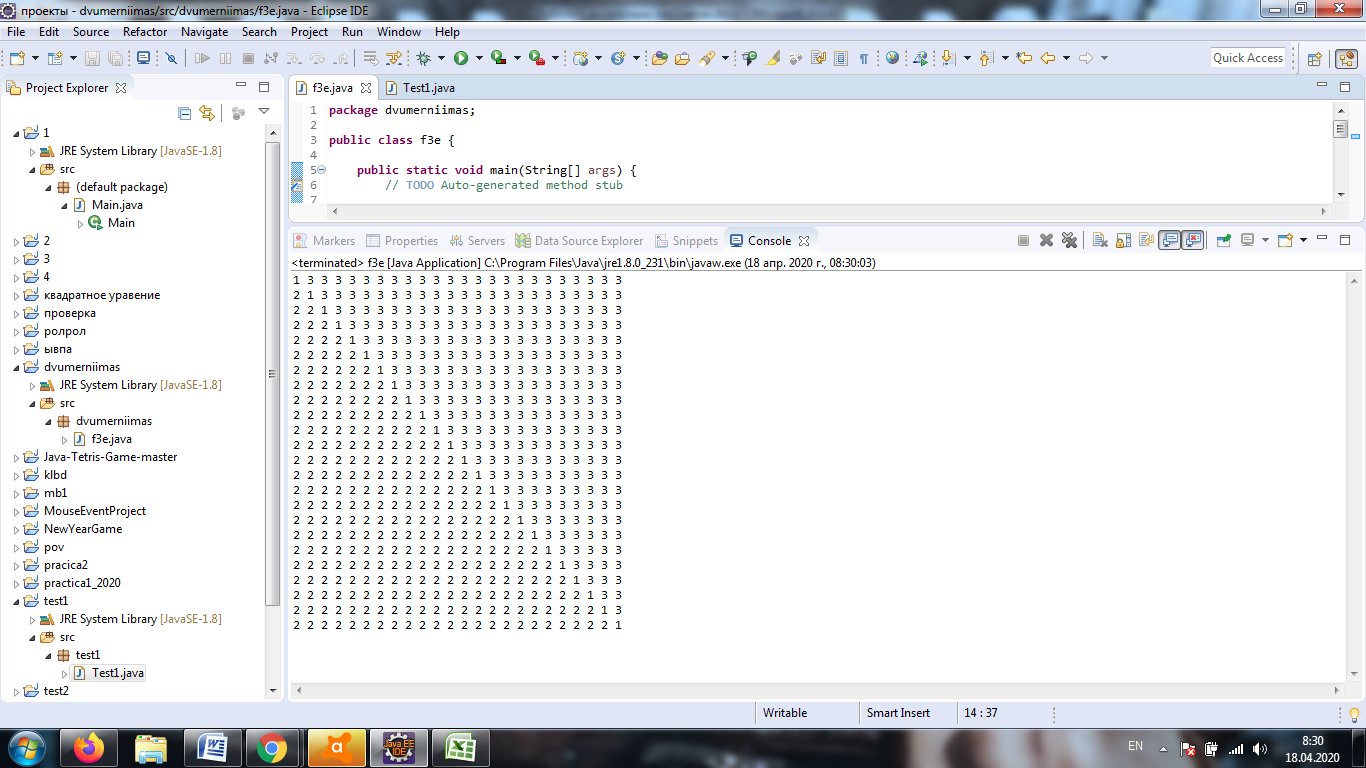
System.***out***.println();

}

2.2. Запустите и посмотрите, что получилось, поменяйте код программы таким образом, чтобы главная диагональ заполнилась единицами, как на картинке (подсказка нужно добавить условие во внутренний цикл сравнивающее значение итераторов цикла):



2.3. Поменяйте код программы таким образом, чтобы все элементы выше главной диагонали были 3, а ниже 2 (подсказка нужно добавить условие во внутренний цикл сравнивающее значение итераторов цикла):



2.4. Поменяйте код программы таким образом, чтобы у вас поучился массив в виде шахматной доски:

