Министерство Образования и Науки Украины

Донецкий Национальный Технический Университет

Кафедра: Прикладной Математики и Информатики

Лабораторная работа № 5

По предмету «Введение в программирование .NET и Java»

Тема**:** «Исключения в Java»

Подготовил:

Лысенко А.С.

Группа: ПИ-13б

Проверил:

асс. каф. ПМИ Павловский Евгений Викторович

Красноармейск 2015

Задание на лабораторную работу:

1. В предыдущих лабораторных работах (№1, 2, 3) добавить контроль правильности исходных данных (как на этапе получения входных данных так и на этапе обработки).
2. При получении неправильных данных генерировать требуемое исключение.
3. Для выполнения лабораторной работы создать свой класс исключений.

Вариант № 7

Код на Java

**package** MyPackage;

**import** java.util.\*;

**class** JustExc **extends** Exception

{

**private** **int** a;

JustExc(**int** c)

{

a = c;

}

}

**public** **class** JavaLab1

{

**public** **static** **void** main(String args[]) **throws** JustExc

{

**int** c = 0;

**int** x = 0;

**int** y = 0;

**int** z = 0;

**int** q = 0;

**int** p = 0;

Scanner scn = **new** Scanner(System.***in***);

**while**(x == 0 || y == 0 || z == 0 || x < 0 || y < 0 || z < 0)

{

**try**

{

System.***out***.print("Введите число x : ");

x = scn.nextInt();

System.***out***.print("Введите число y : ");

y = scn.nextInt();

System.***out***.print("Введите число z : ");

z = scn.nextInt();

**if**(x < 0 || y < 0 || z < 0 || x == 0 || y == 0 || z == 0) **throw** **new** JustExc(c);

{

p = *sum*(x,y,p);

p = *product*(x,y,p);

p = *decrement*(x, p);

p = *leftshift*(x,y);

p = *isand*(x,y);

p = *is*(x,y);

p = *andlog*(x,y,z);

p = *andand*(x,y);

System.***out***.println("Tест : ");

p = *test*(x,y,z,q);

System.***out***.println("Tест 1 : ");

p = *test1*(x,y,z,q);

}

} **catch** (JustExc e)

{

System.***out***.println("Введите число больше нуля или не нуль!");

}

}

}

**public** **static** **int** test(**int** x, **int** y, **int** z, **int** q)

{

q = *plusminus*(x,y,z,q);

q = *minusplus*(x,y,z,q);

q = *umndel*(x,y,z,q);

q = *delumn*(x,y,z,q);

q = *procumn*(x,y,z,q);

**return** q;

}

**public** **static** **int** test1(**int** x, **int** y, **int** z, **int** q)

{

q = *plusplus*(x,y,z,q);

q = *minusminus*(x,y,z,q);

q = *umnumn*(x,y,z,q);

q = *deldel*(x,y,z,q);

q = *procproc*(x,y,z,q);

**return** q;

}

**public** **static** **int** sum(**int** x, **int** y, **int** p)

{

System.***out***.println("Сумма : ");

p = x + y;

System.***out***.println("x + y = " + p);

**return** p;

}

**public** **static** **int** product(**int** x, **int** y, **int** p)

{

System.***out***.println("Произведение : ");

p = x \* y;

System.***out***.println("x \* y = " + p);

**return** p;

}

**public** **static** **int** decrement(**int** x, **int** p)

{

System.***out***.println("Декремент(x) : ");

p = --x;

System.***out***.println("--x = " + p);

**return** p;

}

**public** **static** **int** leftshift(**int** x, **int** y)

{

System.***out***.println("Сдвиг влево(x) << :");

String value = Integer.*toBinaryString*(x);

System.***out***.println("В двоичной : " + value + "\t" + "В десятичной : " + x);

String value1 = Integer.*toBinaryString*(x<<1);

**int** d = 0;

d = x<<1;

System.***out***.println("В двоичной : " + value1 + "\t" + "В десятичной : " + d);

**return** x;

}

**public** **static** **int** isand(**int** x, **int** y)

{

System.***out***.println("Операция(y) &= :");

String value = Integer.*toBinaryString*(y);

System.***out***.println("В двоичной : " + value + "\t" + "В десятичной : " + y);

y &= x; // y = y & x

String value1 = Integer.*toBinaryString*(y&=x);

System.***out***.println("В двоичной : " + value1 + "\t" + "В десятичной : " + y);

**return** y;

}

**public** **static** **int** is(**int** x, **int** y)

{

System.***out***.println("Операция == :");

**if**(x == y)

System.***out***.println("Числа x и y равные");

**else**

System.***out***.println("Числа x и y не равные");

**return** 0;

}

**public** **static** **int** andlog(**int** x, **int** y, **int** z)

{

System.***out***.println("Операция &(лог) : ");

String value = Integer.*toBinaryString*(x);

System.***out***.println("В двоичной : " + value + "\t" + "В десятичной : " + x);

String value1 = Integer.*toBinaryString*(y);

System.***out***.println("В двоичной : " + value1 + "\t" + "В десятичной : " + y);

z = x & y;

String value2 = Integer.*toBinaryString*(z);

System.***out***.println("В двоичной : " + value2 + "\t" + "В десятичной : " + z);

**return** 0;

}

**public** **static** **int** andand(**int** x, **int** y)

{

System.***out***.println("Операция && : ");

System.***out***.println("Число x : " + x);

System.***out***.println("Число y : " + y);

**if** (x > 0 && y > 0)

{

System.***out***.println("x и y больше нуля");

}

**return** 0;

}

**public** **static** **int** plusminus(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Сумма - вычитание :");

q = x + y - z;

System.***out***.println(x + " + " + y + " - " + z + " = " + q);

**return** q;

}

**public** **static** **int** minusplus(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Вычитание - сумма :");

q = x - y + z;

System.***out***.println(x + " - " + y + " + " + z + " = " + q);

**return** q;

}

**public** **static** **int** umndel(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Умножение - деление :");

q = x \* y / z;

System.***out***.println(x + " \* " + y + " / " + z + " = " + q);

**return** q;

}

**public** **static** **int** delumn(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Деление - умножение :");

q = x / y \* z;

System.***out***.println(x + " / " + y + " \* " + z + " = " + q);

**return** q;

}

**public** **static** **int** procumn(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Остаток - умножение :");

q = x % y \* z;

System.***out***.println(x + " % " + y + " \* " + z + " = " + q);

**return** q;

}

**public** **static** **int** plusplus(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Сумма - сумма :");

q = x + y + z;

System.***out***.println(x + " + " + y + " + " + z + " = " + q);

**return** q;

}

**public** **static** **int** minusminus(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Вычитание - вычитание :");

q = x - y - z;

System.***out***.println(x + " - " + y + " - " + z + " = " + q);

**return** q;

}

**public** **static** **int** umnumn(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Умножение - умножение :");

q = x \* y \* z;

System.***out***.println(x + " \* " + y + " \* " + z + " = " + q);

**return** q;

}

**public** **static** **int** deldel(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Деление - деление :");

q = x / y / z;

System.***out***.println(x + " / " + y + " / " + z + " = " + q);

**return** q;

}

**public** **static** **int** procproc(**int** x, **int** y, **int** z, **int** q)

{

System.***out***.println("Остаток - остаток :");

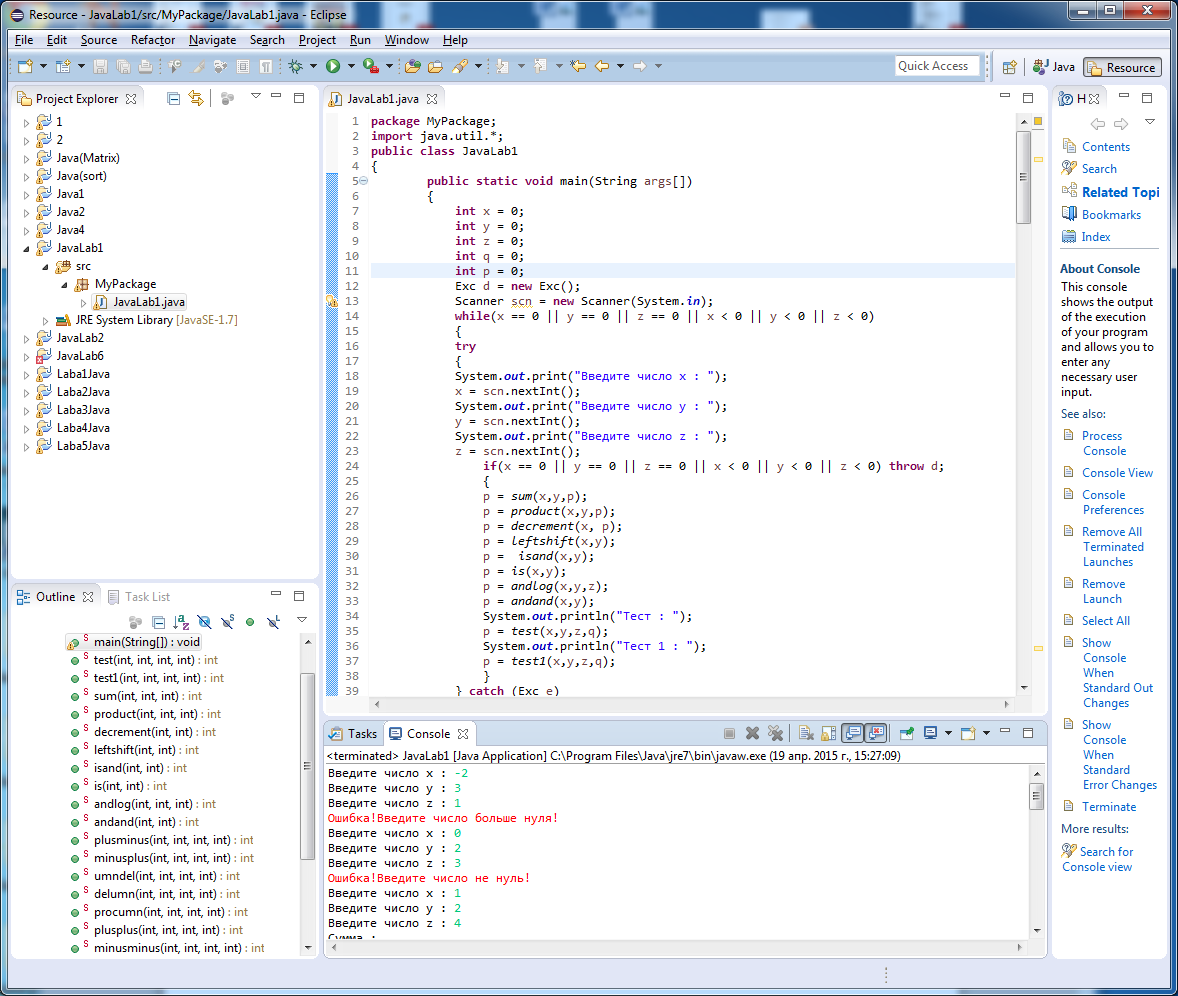
q = x % y % z;

System.***out***.println(x + " % " + y + " % " + z + " = " + q);

**return** q;

}

}



**package** MasModern;

**import** java.util.\*;

**class** JustExc **extends** Exception

{

**private** **int** a;

JustExc(**int** b)

{

a = b;

}

**public** String toString()

{

**return** "JustExc[" + a + "]";

}

}

**class** Matrixxx

{

**static** **int** *n*;

**static** **int** *m*;

**static** **int** *sum*;

**static** **int** *g*;

**static** **int** *k*;

**static** **int** *ch*[][];

**static** **int** *ch1*[];

**static** **boolean** *f* = **false**;

**public** **static** **void** EnterMassive() **throws** JustExc

{

**int** b = 0;

System.***out***.println("Вызван EnterMassive(" + b + ")");

**int** i;

**int** j = 0;

*sum* = 0;

**int** p = 0;

Scanner scn = **new** Scanner(System.***in***);

System.***out***.print("Введите кол-во строк в матрице : ");

*n* = scn.nextInt();

System.***out***.print("Введите кол-во столбцов в матрице : ");

*m* = scn.nextInt();

**if**(*n* <= 0 || *m* <= 0) **throw** **new** JustExc(b);

System.***out***.println("Нормальный выход");

*k* = *n* \* *m*;

//System.out.println(n);

//System.out.println(m);

*ch* = **new** **int** [*n*][*m*];

*ch1* = **new** **int** [*k*];

**for**(i = 0; i < *n*; i++)

{

**for**(j = 0; j < *m*; j++)

{

System.***out***.print("Введите элемент[" + (i + 1) + "][" + (j + 1) + "] матрицы : ");

*ch*[i][j] = scn.nextInt();

}

}

*g* = 0;

**for**(i = 0; i < *n*; i++)

{

**for**(j = 0; j < *m*; j++)

{

*ch1*[*g*] = *ch*[i][j];

*g*++;

}

}

**for**(i = 0; i < *n*; i++)

{

**for**(j = 0 + p; j < *m* - p; j++)

{

*sum*+=*ch*[i][j];

}

p++;

}

}

**public** **static** **void** ShowMassive()

{

**int** i = 0;

**int** j;

System.***out***.println("Ваша матрица : ");

**for**(i = 0; i < *n*; i++)

{

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

**for**(j = 0; j < *m*; j++)

{

System.***out***.print(*ch*[i][j] + "\t");

}

}

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

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

System.***out***.println("Ваша сумма : " + *sum*);

/\*for(g = 0; g < k; g++)

{

System.out.print(ch1[g] + "\t");

}\*/

}

**public** **static** **void** SortMassive()

{

**int** i;

**int** j;

**int** temp = 0;

**while**(!*f*)

{

*f* = **true**;

**for**(*g* = 0; *g* < *k* - 1; *g*++)

{

**if**(*ch1*[*g*] > *ch1*[*g* + 1])

{

temp = *ch1*[*g*];

*ch1*[*g*] = *ch1*[*g* + 1];

*ch1*[*g* + 1] = temp;

*f* = **false**;

}

}

}

/\*for(g = 0; g < k; g++)

{

System.out.print(ch1[g] + "\t");

}\*/

*g* = 0;

**for**(i = 0; i < *n*; i++)

{

**for**(j = 0; j < *m*; j++)

{

*ch*[i][j] = *ch1*[*g*];

*g*++;

}

}

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

**for**(i = 0; i < *n*; i++)

{

**for**(j = 0; j < *m*; j++)

{

System.***out***.print(*ch*[i][j] + "\t");

}

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

}

}

}

**public** **class** MassiveModern

{

**public** **static** **void** main(String[] args)

{

Matrixxx a = **new** Matrixxx();

**try**

{

a.*EnterMassive*();

} **catch** (JustExc e)

{

System.***out***.println("Введите число больше нуля!");

System.***out***.println("Выборошено" + e);

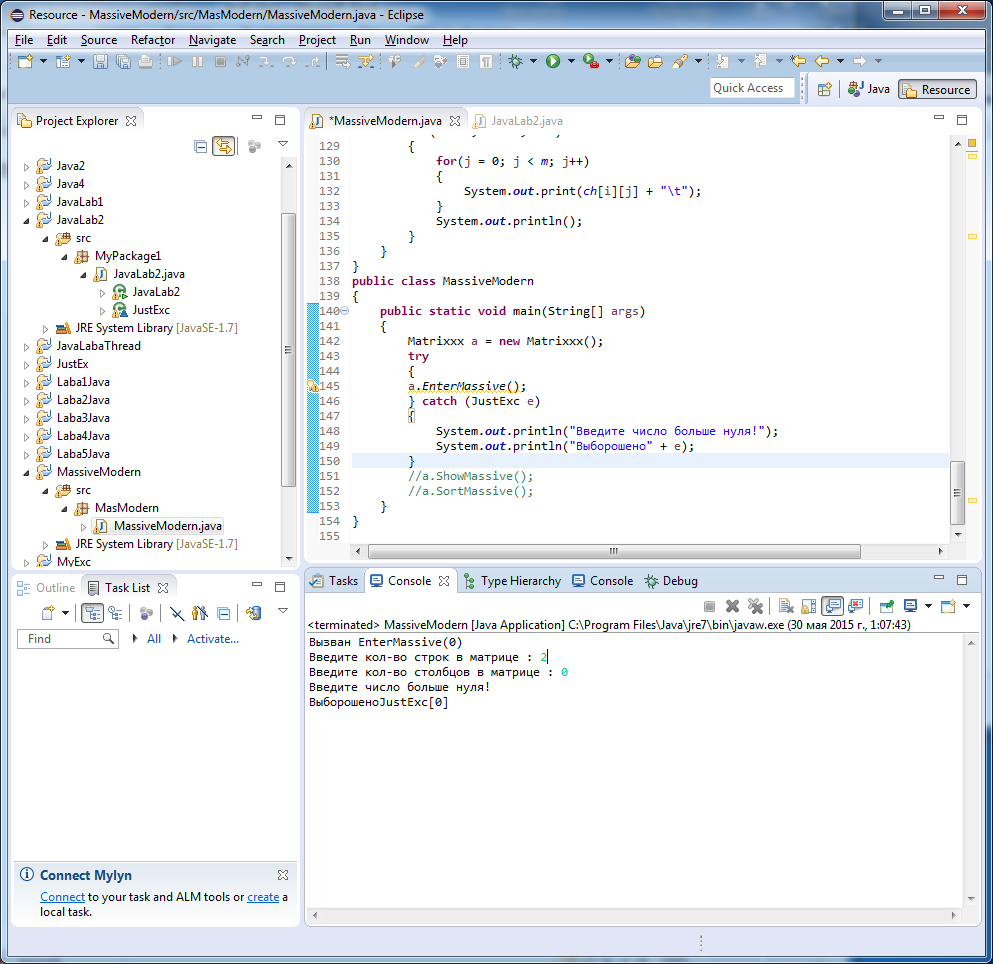
}

a.ShowMassive();

a.SortMassive();

}

}



**package** Pkg4;

**import** java.util.\*;

**class** JustExc **extends** Exception

{

**private** **int** a;

JustExc(**int** c)

{

a = c;

}

}

**public** **class** Java4

{

**public** **static** **void** main(String args[]) **throws** JustExc

{

**int** c = 0;

Scanner scn = **new** Scanner(System.***in***);

String s;

**char** i;

System.***out***.print("Введите строку : ");

s = scn.nextLine();

String ch = s.substring(0,1);

String k = "iCodename";

**while**(k.length() > 1)

{

**try**

{

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

System.***out***.print("Введите символ : ");

k = scn.next();

**if** (k.length() > 1) **throw** **new** JustExc(c);

{

System.***out***.print("Ваша строка : ");

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

System.***out***.print("Первый символ : ");

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

System.***out***.print("Измененная строка : ");

String ch1 = (s.replace(ch, k));

String ch2 = ch1.substring(1, ch1.length());

System.***out***.println(ch + ch2);

}

} **catch**(JustExc e)

{

System.***out***.print("Ошибка!");

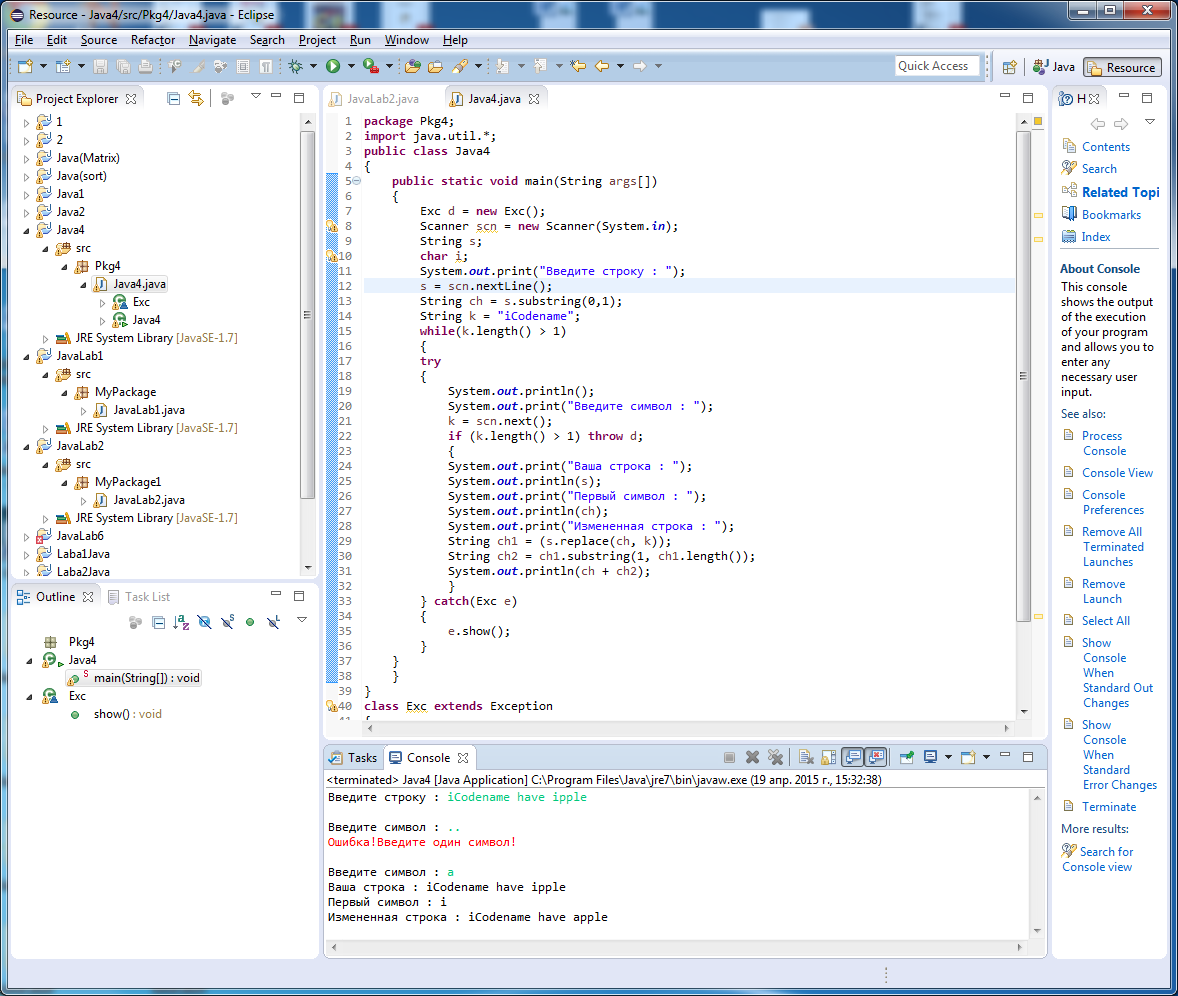
System.***out***.println("Введите один символ!");

}

}

}

}



Ручной просчет

Введите число x : 4

Введите число y : 2

Введите число z : 5

Сумма :

x + y = 6

Произведение :

x \* y = 8

Декремент(x) :

--x = 3

Сдвиг влево << :

В двоичной : 100 В десятичной : 4

В двоичной : 1000 В десятичной : 8

Операция &= :

В двоичной : 10 В десятичной : 2

В двоичной : 0 В десятичной : 0

Операция == :

Числа x и y не равные

Операция &(лог) :

В двоичной : 100 В десятичной : 4

В двоичной : 10 В десятичной : 2

В двоичной : 0 В десятичной : 0

Операция & :

Число x : 4

Число y : 2

x и y больше нуля

Сумма - вычитание :

4 + 2 - 5 = 1

Вычитание - сумма :

4 - 2 + 5 = 7

Умножение - деление :

4 \* 2 / 5 = 1

Деление - умножение :

4 / 2 \* 5 = 10

Остаток - умножение :

4 % 2 \* 5 = 0

Сумма - сумма :

4 + 2 + 5 = 11

Вычитание - вычитание :

4 - 2 - 5 = -3

Умножение - умножение :

4 \* 2 \* 5 = 40

Деление - деление :

4 / 2 / 5 = 0

Остаток - остаток :

4 % 2 % 5 = 0

Введите кол-во строк в матрице : 3

Введите кол-во столбцов в матрице : 4

Введите элемент[1][1]матрицы : 1

Введите элемент[1][2]матрицы : 5

Введите элемент[1][3]матрицы : 2

Введите элемент[1][4]матрицы : 3

Введите элемент[2][2]матрицы : 4

Введите элемент[2][3]матрицы : 5

Ваша матрица :

1 5 2 3

0 4 5 0

0 0 0 0

Ваша сумма : 20

Введите кол-во строк в матрице : 3

Введите кол-во столбцов в матрице : 4

Введите элемент[1][1] : 1

Введите элемент[1][2] : 5

Введите элемент[1][3] : 2

Введите элемент[1][4] : 3

Введите элемент[2][1] : 4

Введите элемент[2][2] : 5

Введите элемент[2][3] : 6

Введите элемент[2][4] : 7

Введите элемент[3][1] : 3

Введите элемент[3][2] : 4

Введите элемент[3][3] : 6

Введите элемент[3][4] : 1

Ваша матрица :

1 5 2 3

4 5 6 7

3 4 6 1

Измененная матрица :

1 1 2 3

3 4 4 5

5 6 6 7

Введите строку : sdfsfdfsdfs

Введите символ : .

Ваша строка : sdfsfdfsdfs

Первый символ : s

Измененная строка : sdf.fdf.df.