**import** java.util.Scanner;

**public** **class** **Skroutz** {

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

// **TODO** Auto-generated method stub

// INPUT

**int** **T**, **N**, **B**, **input**;

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

T = reader.nextInt();

**int**[][] **output** = **new** **int**[T][];

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

N = reader.nextInt();

**while** (N < 1 || N > 100) {

**System**.***out***.print("N is between 1 and 100 only: ");

N = reader.nextInt();

}

B = reader.nextInt();

**while** (B < 1 || B > 32) {

**System**.***out***.print("B is between 1 and 32 only: ");

B = reader.nextInt();

}

output[i] = **new** **int**[((N \* B) / 32) + 1];

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

input = reader.nextInt();

input = input << (B \* j) % 32;

output[i][(j \* B) / 32] = output[i][(j \* B) / 32] | input;

}

}

// OUTPUT

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

**for** (**int** **j** = 0; j < output[i].length; j++) {

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

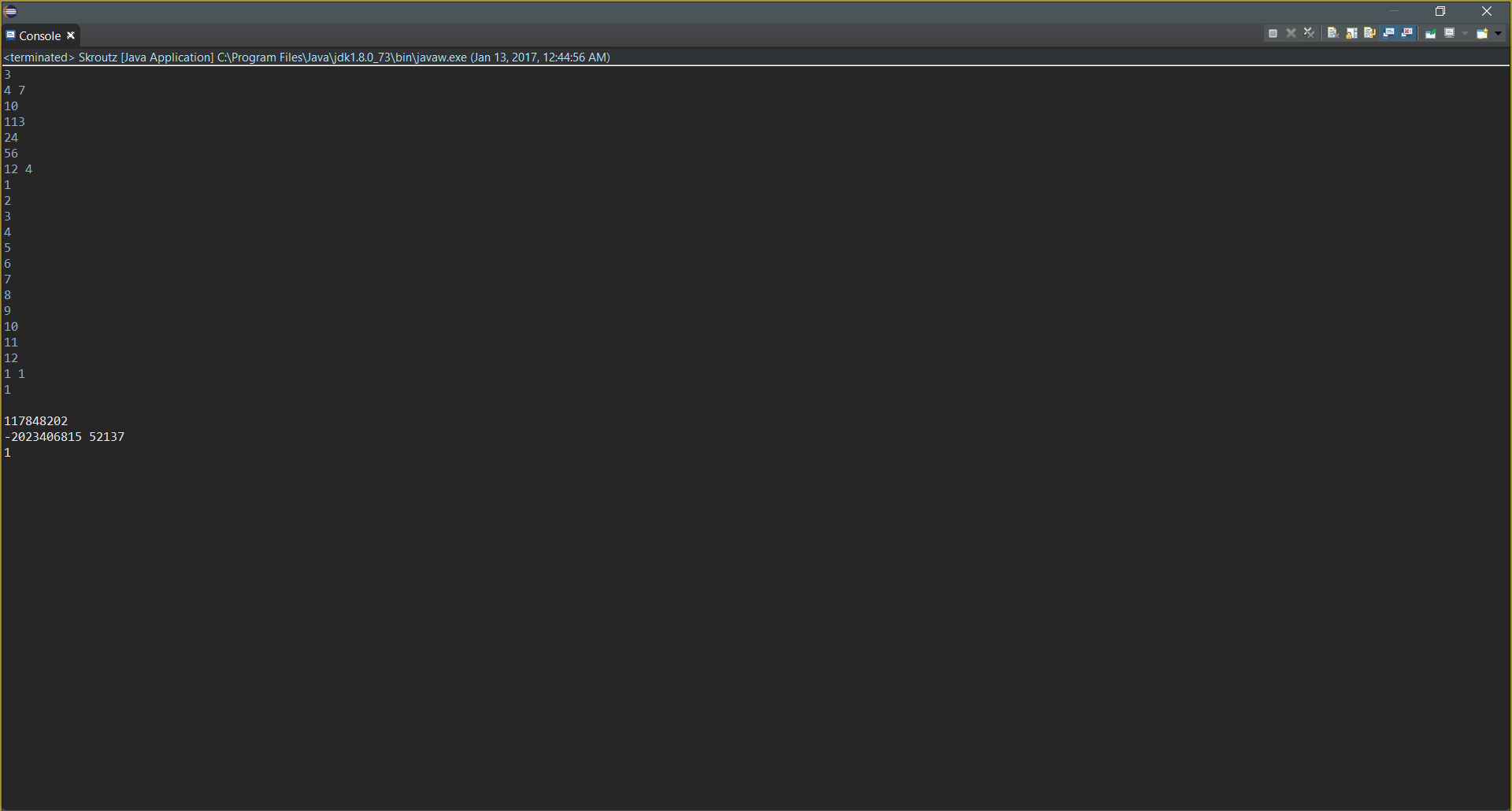
}

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

}

}

}



**import** java.util.Scanner;

**public** **class** **Explosion** {

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

// **TODO** Auto-generated method stub

**int** **N**;

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

N = reader.nextInt();

**while** (N < 1 || N > 100) {

**System**.***out***.print("N is between 1 and 100 only: ");

N = reader.nextInt();

}

**int**[] **T** = **new** **int**[N];

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

T[i] = reader.nextInt();

**while** (T[i] < 0 || T[i] > 1000) {

**System**.***out***.print("T is between 0 and 1000 only: ");

T[i] = reader.nextInt();

}

}

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

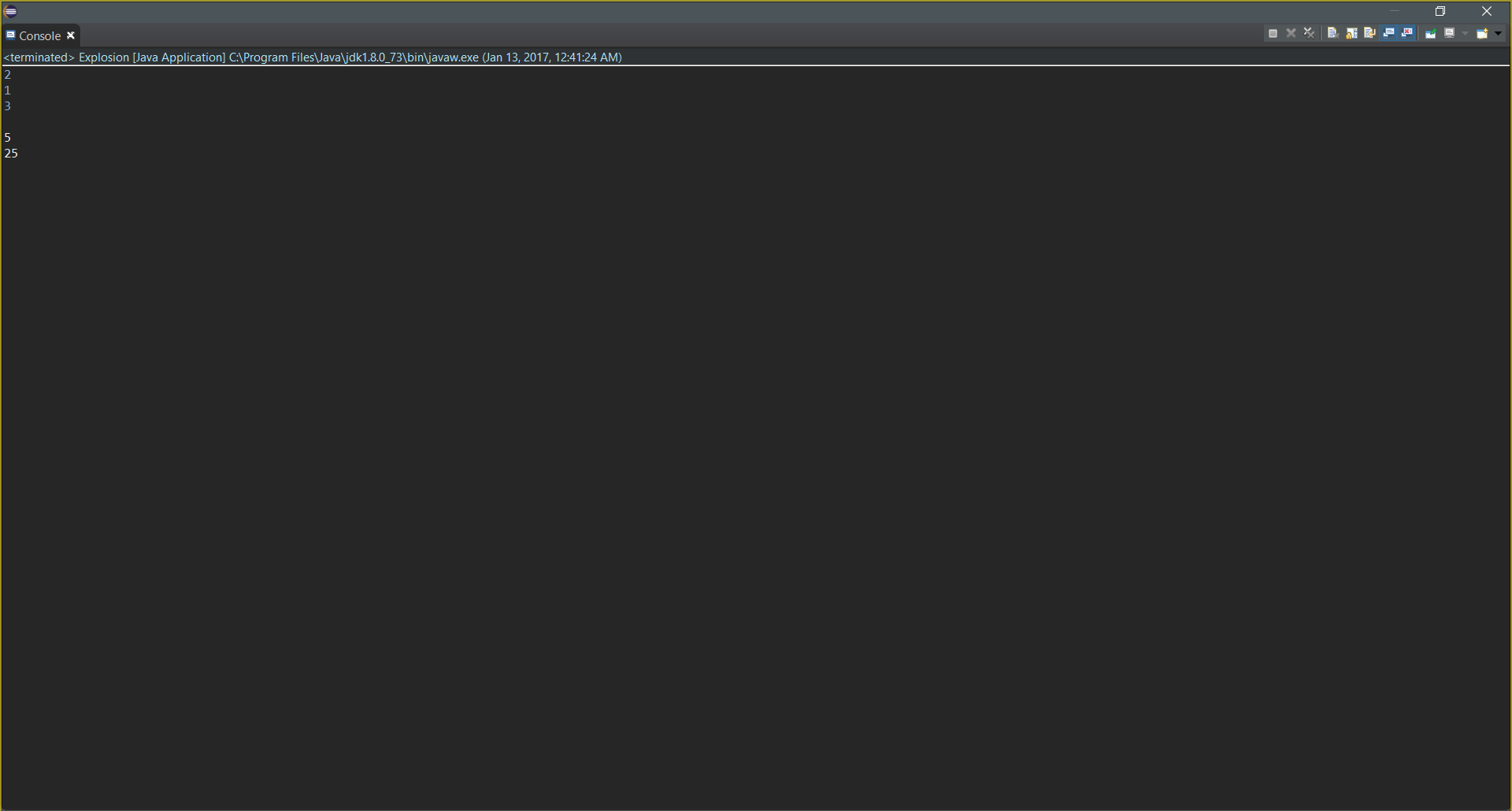
**for** (**int** **j** = 0; j < N; j++) {

**System**.***out***.println(T[j] \* T[j] + (T[j] + 1) \* (T[j] + 1));

}

}

}



**import** java.util.Scanner;

**public** **class** **MatrixMadness** {

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

**int**[][] **matrix** = **new** **int**[4][5];

**for** (**int** **row** = 0; row < 4; row++)

**for** (**int** **col** = 0; col < 5; col++)

matrix[row][col] = row + col + 1;

**int** **N**;

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

N = reader.nextInt();

**while** (N < 1 || N > 100) {

**System**.***out***.print("N is between 1 and 100 only: ");

N = reader.nextInt();

}

// INPUT

**int**[][] **input** = **new** **int**[N][];

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

input[i] = **new** **int**[2];

input[i][0] = reader.nextInt();

**while** (input[i][0] < 1 || input[i][0] > 100) {

**System**.***out***.print("R and C are between 1 and 100 only: ");

input[i][0] = reader.nextInt();

}

input[i][1] = reader.nextInt();

**while** (input[i][1] < 1 || input[i][1] > 100) {

**System**.***out***.print("R and C are between 1 and 100 only: ");

input[i][1] = reader.nextInt();

}

}

// OUTPUT

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

**int** **sum** = 0, **row2**, **col2**;

**for** (**int** **k** = 0; k < N; k++) {

row2 = input[k][0];

col2 = input[k][1];

sum = 0;

**for** (**int** **l** = 0; l < row2; l++) {

**for** (**int** **m** = 0; m < col2; m++) {

sum += matrix[l][m];

}

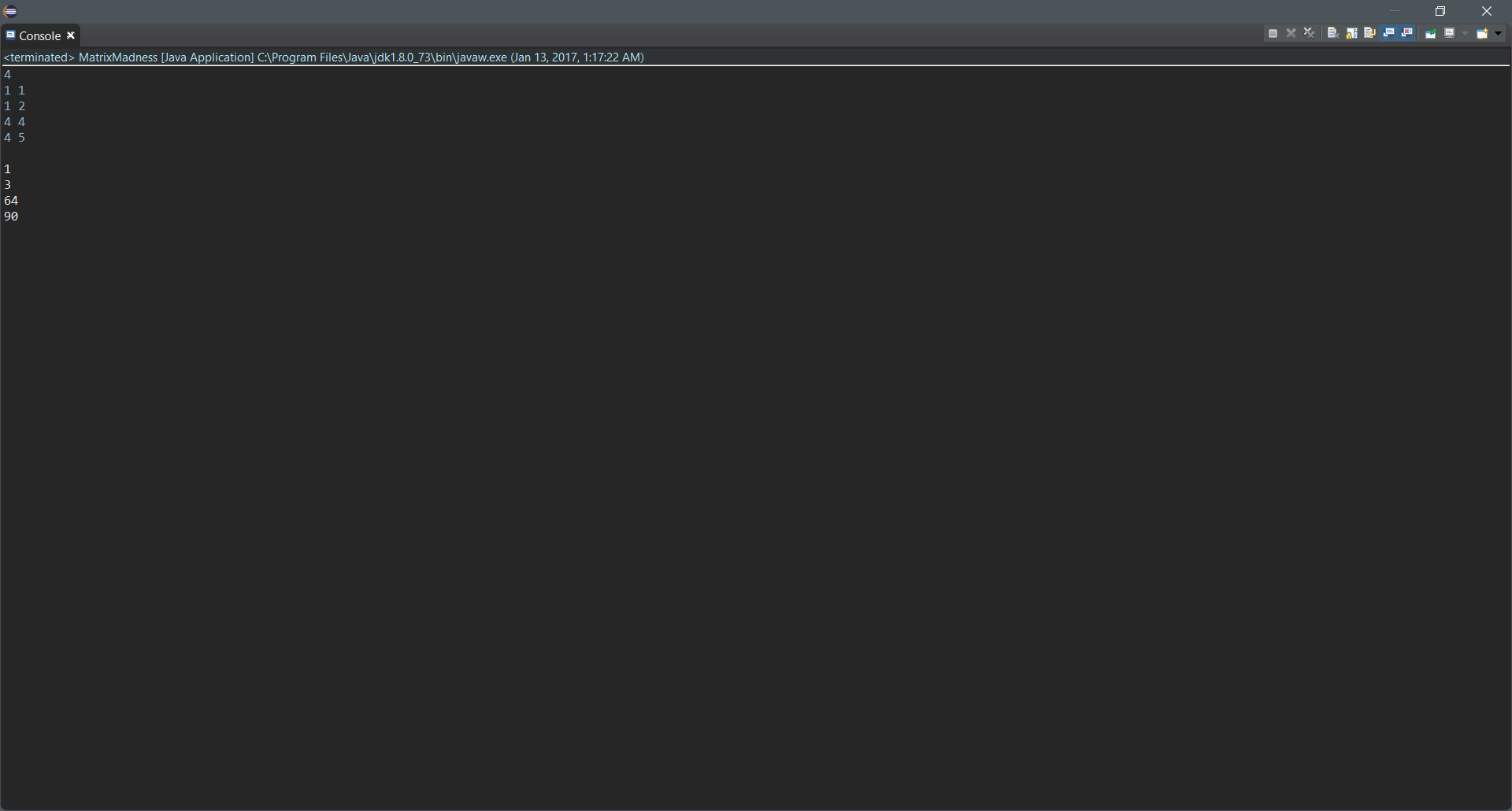
}

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

}

}

}



**import** java.util.Scanner;

**public** **class** **JosephusReturns** {

**static** **int** **getSurvivingPosition**(**int**[] people, **int** kill) {

**int** **count** = people.length;

**int** **skip** = kill - 1;

**int** **skipcount**;

kill--;

**while** (count > 1) {

people[kill] = 0;

count--;

skip++;

skipcount = skip;

kill = (kill + 1) % people.length;

**while** (skipcount != 0 || people[kill] != 1) {

**if** (people[kill] == 1) {

skipcount--;

kill = (kill + 1) % people.length;

} **else**

kill = (kill + 1) % people.length;

}

}

**int** **i** = 0;

**while** (people[i] != 1) {

i++;

}

**return** i + 1;

}

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

// **TODO** Auto-generated method stub

**int** **T**, **N**, **D**;

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

T = reader.nextInt();

// INPUT

**int**[][] **input** = **new** **int**[T][];

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

N = reader.nextInt();

**while** (N < 0 || N > 1000) {

**System**.***out***.println("N is between 0 and 1000 only: ");

N = reader.nextInt();

}

D = reader.nextInt();

**while** (D < 0 || D > 1000) {

**System**.***out***.println("D is between 0 and 1000000 only: ");

D = reader.nextInt();

}

input[i] = **new** **int**[2];

input[i][0] = N;

input[i][1] = D;

}

// OUTPUT

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

**int** **kill**;

**int**[] **people**;

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

people = **new** **int**[input[j][0]];

**for** (**int** **k** = 0; k < people.length; k++)

people[k] = 1;

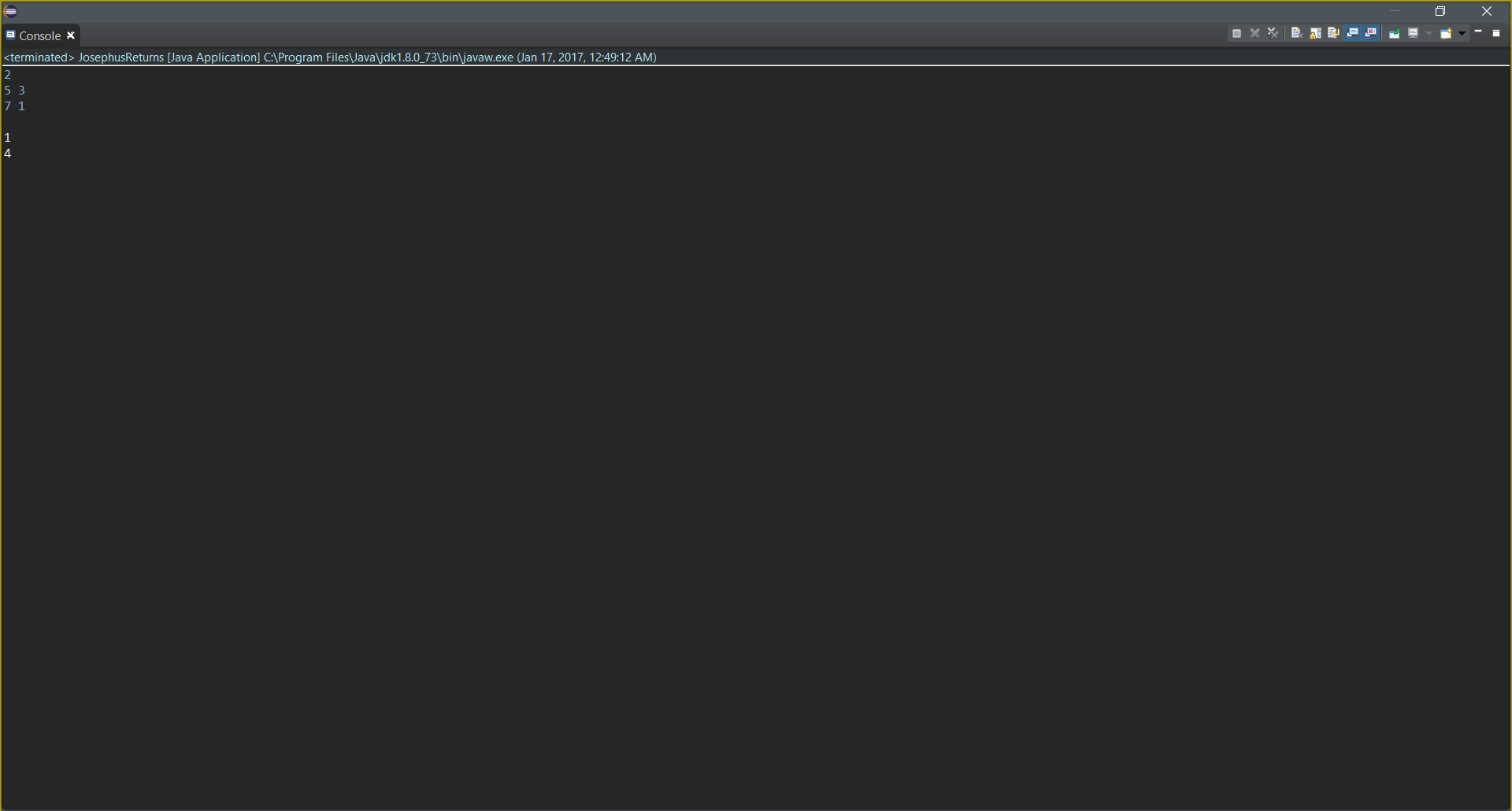
kill = input[j][1];

**System**.***out***.println(*getSurvivingPosition*(people, kill));

}

}

}



**import** java.util.Scanner;

**public** **class** **Reflection** {

**static** **int** **reflectionAcross**(**int**[] coordinate) {

**int** **reflection** = -1;

**if** (coordinate[1] == -coordinate[3]) {

**if** (coordinate[0] == -coordinate[2])

reflection = 3;

**else**

reflection = 1;

} **else** **if** (coordinate[0] == -coordinate[2]) {

**if** (coordinate[1] == -coordinate[3])

reflection = 3;

**else**

reflection = 2;

}

**return** reflection;

}

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

// **TODO** Auto-generated method stub

**int** **T**;

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

T = reader.nextInt();

// INPUT

**int**[][] **input** = **new** **int**[T][];

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

input[i] = **new** **int**[4];

input[i][0] = reader.nextInt();

**while** (input[i][0] < -10000 || input[i][0] > 10000) {

**System**.***out***.println("Value is between -10000 and 10000 only: ");

input[i][0] = reader.nextInt();

}

input[i][1] = reader.nextInt();

**while** (input[i][1] < -10000 || input[i][1] > 10000) {

**System**.***out***.println("Value is between -10000 and 10000 only: ");

input[i][1] = reader.nextInt();

}

input[i][2] = reader.nextInt();

**while** (input[i][2] < -10000 || input[i][2] > 10000) {

**System**.***out***.println("Value is between -10000 and 10000 only: ");

input[i][2] = reader.nextInt();

}

input[i][3] = reader.nextInt();

**while** (input[i][3] < -10000 || input[i][3] > 10000) {

**System**.***out***.println("Value is between -10000 and 10000 only: ");

input[i][3] = reader.nextInt();

}

}

// OUTPUT

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

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

**switch** (*reflectionAcross*(input[j])) {

**case** -1:

**System**.***out***.println("The points are not reflections.");

**break**;

**case** 1:

**System**.***out***.println("The points are reflections across the x-axis.");

**break**;

**case** 2:

**System**.***out***.println("The points are reflections across the y-axis.");

**break**;

**case** 3:

**System**.***out***.println("The points are reflections across the origin.");

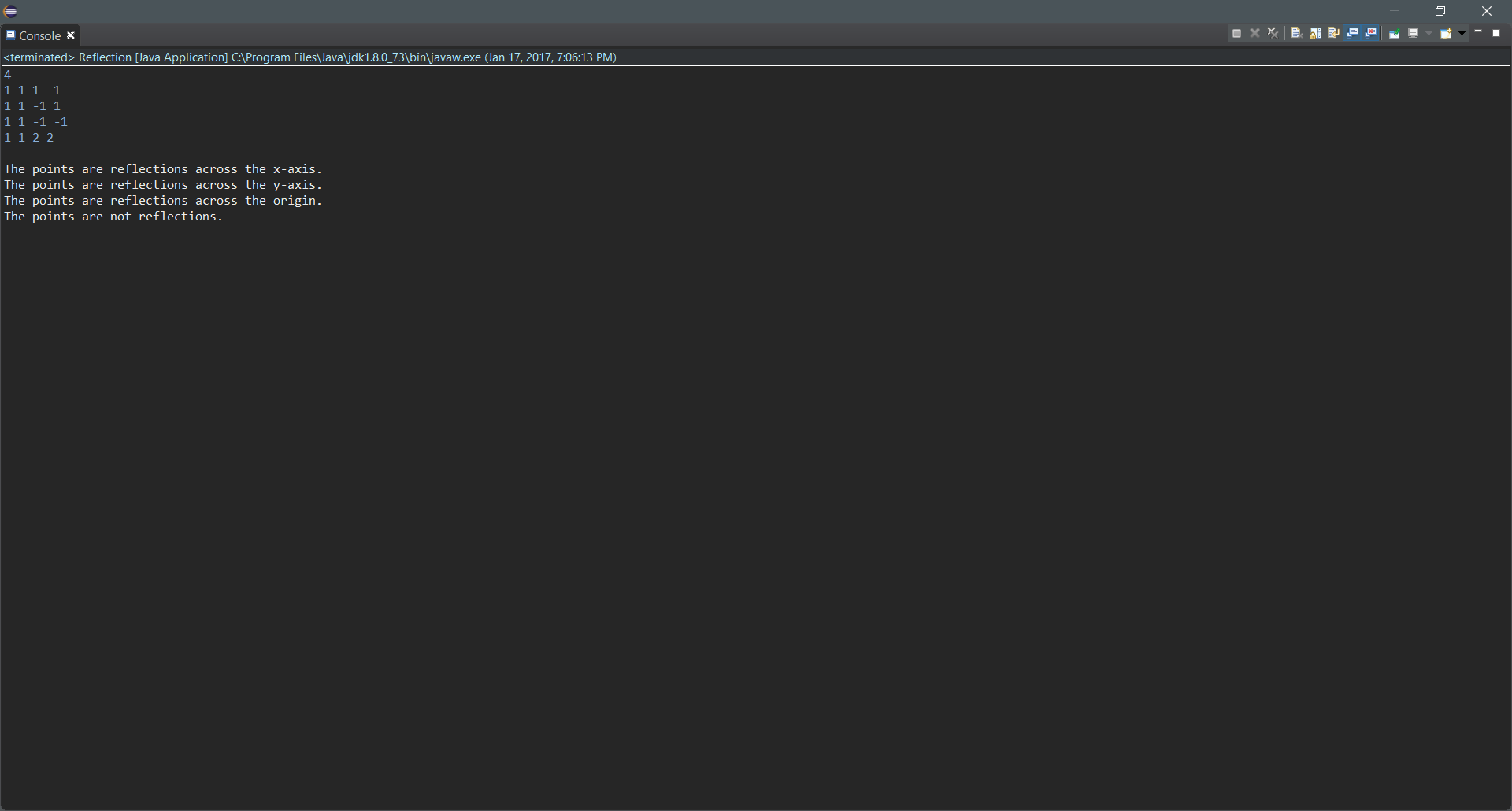
**break**;

}

}

}

}



**import** java.util.Scanner;

**public** **class** **Elegance** {

**static** **String** **elegantTimeLog**(**int** n)

{

**String** **log** = "";

**int** **hour**;

**int** **minute**;

hour = n / 60;

minute = n % 60;

**if** (hour != 0)

log += hour + " hour";

**if** (hour > 1)

log += "s";

**if** (minute != 0)

**if** (hour != 0)

log += ", ";

log += minute + " minute";

**if** (minute > 1)

log += "s";

**return** log;

}

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

// **TODO** Auto-generated method stub

**int** **T**, **n**;

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

T = reader.nextInt();

// INPUT

**int**[] **input** = **new** **int**[T];

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

n = reader.nextInt();

**while** (n < 1 || n > 100000) {

**System**.***out***.println("Value is between 1 and 100000 only: ");

n = reader.nextInt();

}

input[i] = n;

}

// OUTPUT

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

**for** (**int** **j** = 0; j < T; j++)

**System**.***out***.println(*elegantTimeLog*(input[j]));

}

}

