Started on	Thursday, 28 December 2023, 4:37 PM
State	Finished
Completed on	Thursday, 28 December 2023, 5:29 PM
Time taken	51 mins 46 secs

Question **1**

Correct

Marked out of 25.00

Suppandi is trying to take part in the local village math quiz. In the first round, he is asked about shapes and areas. Suppandi, is confused, he was never any good at math. And also, he is bad at remembering the names of shapes. Instead, you will be helping him calculate the area of shapes.

· When he says rectangle he is actually referring to a square.

· When he says square, he is actually referring to a triangle.

· When he says triangle he is referring to a rectangle

· And when he is confused, he just says something random. At this point, all you can do is say 0.

Help Suppandi by printing the correct answer in an integer.

Input Format

· Name of shape (always in upper case R à Rectangle, S à Square, T à Triangle)

· Length of 1 side

· Length of other side

Note: In case of triangle, you can consider the sides as height and length of base

Output Format

· Print the area of the shape.

Sample Input 1

T

10

20

Sample Output 1

200

Sample Input 2

S

30

40

Sample Output 2

600

Sample Input 3

R

10

10

Sample Output 3

100

Sample Input 4

G

8

8

Sample Output 4

0

Sample Input

C

9

10

Sample Output 4

0

Explanation:

· First is output of area of rectangle

· Then, output of area of triangle

· Then output of area square

· Finally, something random, so we print 0

For example:

Input	Result
T 10 20	200
S 30 40	600
B 2 11	0
R 10 30	300

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class shapes
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        char s = sc.next().charAt(0);
        int s1 = sc.nextInt();
        int s2 = sc.nextInt();

        if(s=='R')
        {
            System.out.println(s1*s2);
        }
        else if(s=='T')
        {
            System.out.println(s1*s2);
        }
        else if(s=='S')
        {
            System.out.println((s1*s2)/2);
        }
        else
        {
            System.out.println(0);
        }
    }
}
```

	Input	Expected	Got	
✔	T 10 20	200	200	✔
✔	S 30 40	600	600	✔

	Input	Expected	Got	
✔	B 2 11	0	0	✔
✔	R 10 30	300	300	✔

Passed all tests! ✔

Question **2**

Correct

Marked out of 25.00

Objective

In this challenge, we're getting started with conditional statements.

Task

Given an integer, ***n***, perform the following conditional actions:

· If ***n*** is odd, print Weird

· If ***n*** is even and in the inclusive range of ***2*** to ***5***, print ***Not Weird***

· If ***n*** is even and in the inclusive range of ***6*** to ***20***, print ***Weird***

· If ***n*** is even and greater than ***20***, print ***Not Weird***

Complete the stub code provided in your editor to print whether or not ***n*** is weird.

Input Format

A single line containing a positive integer, ***n***.

Constraints

· $1 \leq n \leq 100$

Output Format

Print Weird if the number is weird; otherwise, print Not Weird.

Sample Input 0

3

Sample Output 0

Weird

Sample Input 1

24

Sample Output 1

Not Weird

Explanation

Sample Case 0: ***n = 3***

n is odd and odd numbers are weird, so we print ***Weird***.

Sample Case 1: ***n = 24***

n > 20 and ***n*** is even, so it isn't weird. Thus, we print ***Not Weird***.

For example:

Input	Result
3	Weird
24	Not Weird

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class Weird
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        if(n%2==1)
        {
            System.out.print("Weird");
        }
        else if((n>=2) && (n<=5))
        {
            System.out.print("Not Weird");
```

```
        }
        else if((n>=6) && (n<=20))
        {
            System.out.print("Weird");
        }
        else
        {
            System.out.print("Not Weird");
        }
    }
}
```

	Input	Expected	Got	
✔	3	Weird	Weird	✔
✔	24	Not Weird	Not Weird	✔

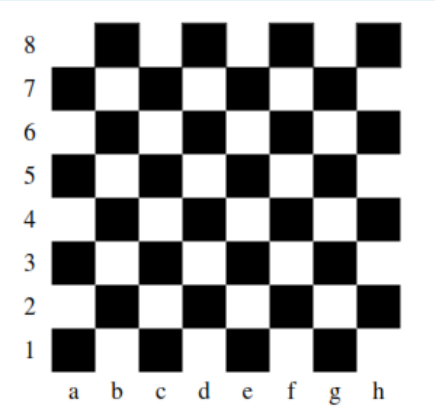
Passed all tests! ✔

Question **3**

Correct

Marked out of 25.00

Positions on a chess board are identified by a letter and a number. The letter identifies the column, while the number identifies the row, as shown below:



Write a program that reads a position from the user. Use an if statement to determine if the column begins with a black square or a white square. Then use modular arithmetic to report the color of the square in that row. For example, if the user enters a1 then your program should report that the square is black. If the user enters d5 then your program should report that the square is white. Your program may assume that a valid position will always be entered. It does not need to perform any error checking.

Sample Input 1

a 1

Sample Output 1

The square is black.

Sample Input 2

d 5

Sample Output 2

The square is white.

For example:

Input	Result
a 1	The square is black.
d 5	The square is white.

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class Chess
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        char l = sc.next().charAt(0);
        int n = sc.nextInt();

        if((l=='a')||(l=='c')||(l=='e')||(l=='g'))
        {
            if(n%2==1)
            {
                System.out.println("The square is black.");
            }
            else
            {
                System.out.println("The square is white.");
            }
        }
        else
        {
            if(n%2==0)
```

```
23              if(n%2==0)
24              {
25                  System.out.println("The square is black.");
26              }
27              else
28              {
29                  System.out.println("The square is white.");
30              }
31          }
32
33      }
34  }
```

	Input	Expected	Got	
✔	a 1	The square is black.	The square is black.	✔
✔	d 5	The square is white.	The square is white.	✔

Passed all tests! ✔

Question **4**

Incorrect

Marked out of 25.00

Write a program that determines the name of a shape from its number of sides. Read the number of sides from the user and then report the appropriate name as part of a meaningful message. Your program should support shapes with anywhere from 3 up to (and including) 10 sides. If a number of sides outside of this range is entered then your program should display an appropriate error message.

Sample Input 1

3

Sample Output 1

Triangle

Sample Input 2

7

Sample Output 2

Heptagon

Sample Input 3

11

Sample Output 3

The number of sides is not supported.

For example:

Input	Result
3	Triangle
7	Heptagon
11	The number of sides is not supported.

Answer: (penalty regime: 0 %)

```
import java.util.*;
public class ShapeBySides
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int sides = sc.nextInt();
        switch(sides)
        {
            case 3:
                System.out.println("Triangle");
                break;
            case 4:
                System.out.println("Square");
                break;
            case 5:
                System.out.println("Pentagon");
                break;
            case 6:
                System.out.println("Hexagon");
                break;
            case 7:
                System.out.println("Heptagon");
                break;
            case 8:
                System.out.println("Octagon");
                break;
            case 9:
                System.out.println("Nanogon");
                break;
            case 10:
                System.out.println("Septagon");
                break;
            default:
```

```
            default:
                System.out.println("The number of sides is not supported.");
        }
    }
}
```

	Input	Expected	Got	
✔	3	Triangle	Triangle	✔
✔	7	Heptagon	Heptagon	✔
✔	11	The number of sides is not supported.	The number of sides is not supported.	✔

Your code failed one or more hidden tests.
Your code must pass all tests to earn any marks. Try again.