

**Paranidharan.R**

**ECE - D**

**240801238**

**ProblemStatement1:**

**Write a program to read two integer values and print true if both the numbers end with the same digit, otherwise print false. Example: If 698 and 768 are given, program should print true as they both end with 8.**

**SampleInput1**

**2553**

**SampleOutput1**

**false**

**SampleInput2**

**2777**

**SampleOutput2**

---

---

True

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     int a,b;
4     scanf("%d %d",&a,&b);
5     if(a%10==b%10){
6         printf("true");
7     }
8 }
9 else{
10     printf("false");
11 }
12 }
13 return 0;
14 }
```

	Input	Expected	Got	
✓	25 53	false	false	✓
✓	27 77	true	true	✓

Passed all tests! ✓

## ProblemStatement2:

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

- $1 < n < 100$

### Output Format

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

### Sample Input 0

3

### Sample Output 0

Weird

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     int n;
4     scanf("%d",&n);
5     if(n%2!=0){
6         printf("Weird\n");
7     }
8 }
9 else{
10     if(n>=2 && n<=5){
11         printf("Not Weird\n");
12     }
13     else if(n>=6 && n<=20){
14         printf("Weird\n");
15     }
16     else if(n>20){
17         printf("Not Weird\n");
18     }
19 }
20 return 0;
21 }
```

	Input	Expected	Got	
✓	3	Weird	Weird	✓
✓	24	Not Weird	Not Weird	✓

Passed all tests! ✓

### ProblemStatement3:

ThreenumbersformaPythagoreantripleifthesumofsquaresoftwo numbersisequaltothesquareofthethird.Forexample,3,5and4forma Pythagoreantriple,since $3*3+4*4=25=5*5$ Youaregiventhreeintegers, a,b,andc.Theyneednotbegiveninincreasingorder.Iftheyforma Pythagoreantriple,thenprint"yes",otherwise,print"no".Pleasenotethat theoutputmessageisinsmallletters.

### SampleInput1

3

5

4

SampleOutput1

Yes

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     int a,b,c;
4     scanf("%d %d %d",&a,&b,&c);
5     int largest =a>b?(a>c?a:c):(b>c?b:c);
6     int sum_square=a*a+b*b+c*c-largest*largest;
7     if(sum_square==largest*largest){
8         printf("yes\n");
9     }
10    else{
11        printf("no\n");
12    }
13    return 0;
14 }
```

	Input	Expected	Got	
✓	3 5 4	yes	yes	✓
✓	5 8 2	no	no	✓

Passed all tests! ✓

ProblemStatement4:

---

**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.**

**Answer:** (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     int side;
4     scanf("%d",&side);
5     if(side>=3 && side<=10){
6         switch (side){
7             case 3:
8                 printf("Triangle\n");
9                 break;
10
11             case 4:
12                 printf("Quadrilateral\n");
13                 break;
14             case 5:
15                 printf("Pentagon\n");
16                 break;
17             case 6:
18                 printf("Hexagon\n");
19                 break;
20             case 7:
21                 printf("Heptagon\n");
22                 break;
23             case 8:
24                 printf("Octogon");
25                 break;
26             case 9:
27                 printf("Nonagon");
28                 break;
29             case 10:
30                 printf("Decagon");
31                 break;
32         }
33     }
```

```

32     }
33 }
34 else{
35     printf("The number of sides is not supported.\n");
36 }
37 return 0;
38 }

```

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

Passed all tests! ✓

## ProblemStatement5:

TheChinesezodiacassignsanimalstoyearsina12-yearcycle.One12-year cycleisshowninthetablebelow.Thepatternrepeatsfromthere,with 2012beinganotheryearoftheDragon,and1999beinganotheryearofthe Hare.

**YearAnimal**

**2000 Dragon**

**2001Snake**

**2002Horse**

**2003Sheep**

**2004Monkey**



---

**2005Rooster**

**2006Dog**

**2007Pig**

**2008Rat**

**2009Ox**

**2010Tiger**

**2011Hare**

**Writeaprogramthatreadsayearfromtheuseranddisplaystheanimal associatedwiththatyear.Yourprogramshouldworkcorrectlyforanyyear greaterthanorequaltozero,notjusttheoneslistedinthetable.**

**SampleInput1**

**2004**

**SampleOutput1 Monkey**

**SampleInput2**

**2010**

## SampleOutput2

### Tiger

Answer: (penalty regime: 0 %)

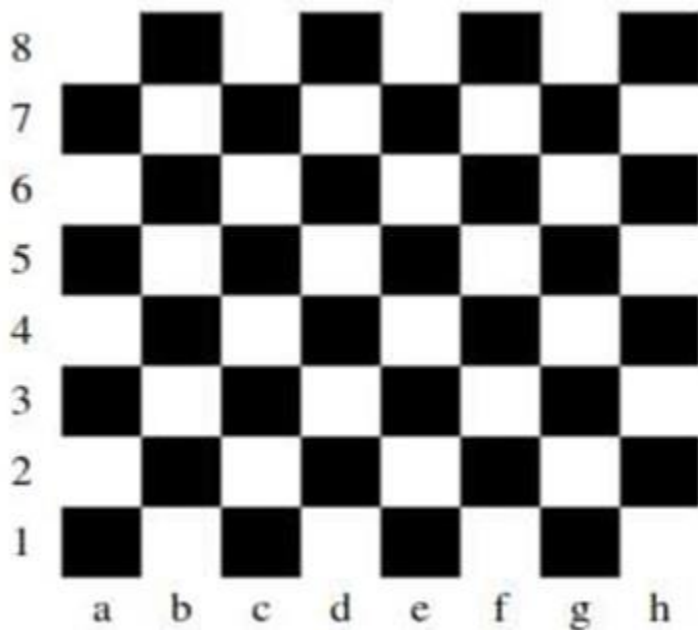
```
1 #include<stdio.h>
2 int main(){
3     int year;
4     char*animals[]={ "Dragon","Snake","Horse","Sheep","Monkey","Roaster","Dog","Pig","Rat","Ox","Tiger","Hare"};
5     scanf("%d",&year);
6     int index=(year-2000)%12;
7     printf("%s\n",animals[index]);
8     return 0;
9 }
10 }
```

	Input	Expected	Got	
✓	2004	Monkey	Monkey	✓
✓	2010	Tiger	Tiger	✓

Passed all tests! ✓

## ProblemStatement6:

Positionsonachessboardareidentifiedbyaletterandanumber.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 many error checking.

**Sample Input 1**

**a1**

**Sample Output 1** The

square is black.

## SampleInput2

d5

## SampleOutput2 The

square is white.

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 #include<ctype.h>
3 int main(){
4     char column,row;
5     scanf("%c %c",&column,&row);
6     column=tolower(column);
7     int intial_color=(column=='a' || column=='h')?1:0;
8     int square_color=(intial_color+(row-'1'))%2;
9     if(square_color==1){
10         printf("The square is black.\n");
11     }
12     else{
13         printf("The square is white.\n");
14     }
15     return 0;
16 }
17
```

	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! ✓

## ProblemStatement7:

---

**Somedatasetsspecifydatesusingtheyearanddayofyearratherthantheyear,month,anddayofmonth.Thedayofyear(DOY)isthesequentialday numberstartingwithday1onJanuary1st.Therearetwocalendars-one fornormalyearswith365days,andoneforleapyearswith366days.Leap yearsaredivisibleby4.Centuries,like1900,arenotleapyearsunlessthey aredivisibleby400.So,2000wasaleapyear.Tofindthedayofyear numberforastandarddate,scandowntheJancolumntofindthedayof month,thenscanacrosstotheappropriatemonthcolumnandreadtheday ofyearnumber.Reversetheprocesstofindthestandarddateforagiven dayofyear.WriteaprogramtoprinttheDayofYearofagivendate,month andyear.**

**SampleInput1**

**18**

**6**

**2020**

**SampleOutput1**

**170**

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     int date,month,year;
4     int days[]={31,28,31,30,31,30,31,31,30,31,30,31};
5     scanf("%d %d %d",&date,&month,&year);
6     if((year%4==0&&year%100!=0)|| (year%400==0)){
7         days[1]=29;
8     }
9     int DOT=date;
10    for(int i=0;i<(month-1);i++){
11        DOT += days[i];
12    }
13    printf("%d\n",DOT);
14    return 0;
15 }
```

	Input	Expected	Got	
✓	18 6 2020	170	170	✓

Passed all tests! ✓

## ProblemStatement8:

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 uppercase R-->Rectangle, S-->Square, T-->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

## SampleOutput1

200

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     char shape;
4     int side1,side2;
5     scanf("%c %d %d",&shape,&side1,&side2);
6     int area;
7     switch(shape){
8         case 'R':
9             area=side1*side2;
10            break;
11         case 'S':
12             area=(side1*side2)/2;
13            break;
14         case 'T':
15             area=side1*side2;
16            break;
17         default:
18             area=0;
19     }
20     printf("%d\n",area);
21     return 0;
22 }
```

	Input	Expected	Got	
✓	T 10 20	200	200	✓
✓	S 30 40	600	600	✓
✓	B 2 11	0	0	✓
✓	R 10 30	300	300	✓
✓	S 40 50	1000	1000	✓

Passed all tests! ✓



---

### **ProblemStatement9:**

**Superman is planning a journey to his home planet. It is very important for him to know which day he arrives there. They don't follow the 7-day week like us. Instead, they follow a 10-day week with the following days:**

**DayNumberNameofDay**

**1 Sunday**

**2 Monday**

**3 Tuesday**

**4 Wednesday**

**5 Thursday**

**6 Friday**

**7 Saturday**

**8 Kryptonday**

**9 Coluday**

**10 Daxamday**

**Here are the rules of the calendar:**

---

- The calendar starts with Sunday always.

- It has only 296 days. After the 296th day, it goes back to Sunday.

You begin your journey on a Sunday and will reach after  $n$ . You have to tell on which day you will arrive when you reach there.

**Input format:**

- Contains a number  $n$  ( $0 < n$ )

**Output format:**

Print the name of the day you are arriving on

**Sample Input**

**7**

**Sample Output**

**Kryptonday**

**Sample Input**

**1**

## SampleOutput

### Monday

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     int n;
4     const char *days[]={"Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday","Kryptonday","Coluday"};
5     scanf("%d",&n);
6     int arrival_day=(n%296)%10;
7     printf("%s\n",days[arrival_day]);
8     return 0;
9 }
```

	Input	Expected	Got	
✓	7	Kryptonday	Kryptonday	✓
✓	1	Monday	Monday	✓

Passed all tests! ✓