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. Sample Input 1 25 53 Sample Output 1 false Sample Input 2 27 77 Sample Output 2 true

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int x,y;
    scanf("%d %d",&x,&y);
    if(x%10==y%10)
    {
        printf("true");
    }
    else
    {
        printf("false");
    }
    return 0;
}
```

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

Passed all tests! ✓

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.

```c
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    if(n%2==0){
        if(n>=2 && n<=5){
            printf("Not Weird");}
        if(n>=6 && n<=20){
            printf("Weird");}
        if(n>20){
            printf("Not Weird");}}
        else{
            printf("Weird");}

```

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

Passed all tests! ✓

Three numbers form a Pythagorean triple if the sum of squares of two numbers is equal to the square of the third. For example, 3, 5 and 4 form a Pythagorean triple, since 3*3 + 4*4 = 25 = 5*5 You are given three integers, a, b, and c. They need not be given in increasing order. If they form a Pythagorean triple, then print "yes", otherwise, print "no". Please note that the output message is in small letters. Sample Input 1 3 5 4 Sample Output 1 yes Sample Input 2 5 8 2 Sample Output 2 no

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
```

```c
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if(a*a+b*b==c*c)
    {
        printf("yes");
    }
    else if(a*a+c*c==b*b)
    {
        printf("yes");
    }
    else if (b*b+c*c==a*a)
    {
```

```
    {
        printf("yes");
    }
    else if (b*b+c*c==a*a)
    {
        printf("yes");
    }
    else
    {
        printf("no");
    }
    return 0;
}
```

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

Passed all tests! ✓

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

```c
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    if(n==3)
    {
        printf("Triangle");
    }
    else if(n==4)
    {
        printf("Square");
    }
    else if(n==5)
    {
```

```
14      else if(n==5)
15      {
16          printf("Pentagon");
17      }
18      else if(n==6)
19      {
20          printf("Hexagon");
21      }
22      else if(n==7)
23      {
24          printf("Heptagon");
25      }
26      else if(n==8)
27      {
28          printf("Octagon");
```

```
    }
    else if(n==9)
    {
        printf("Nonagon");
    }
    else if(n==10)
    {
        printf("Decagon");
    }
    else
    {
        printf("The number of sides is not supported.");
    }
    return 0;
}
```

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

Passed all tests! ✓

The Chinese zodiac assigns animals to years in a 12-year cycle. One 12-year cycle is shown in the table below. The pattern repeats from there, with 2012 being another year of the Dragon, and 1999 being another year of the Hare.

Year	Animal
2000	Dragon
2001	Snake

```c
#include<stdio.h>
int main()
{
    int year;
    scanf("%d",&year);
    if(year%12==8)
    {
        printf("Dragon");
    }
    else if(year%12==9)
    {
        printf("Snake");
    }
    else if(year%12==10)
```

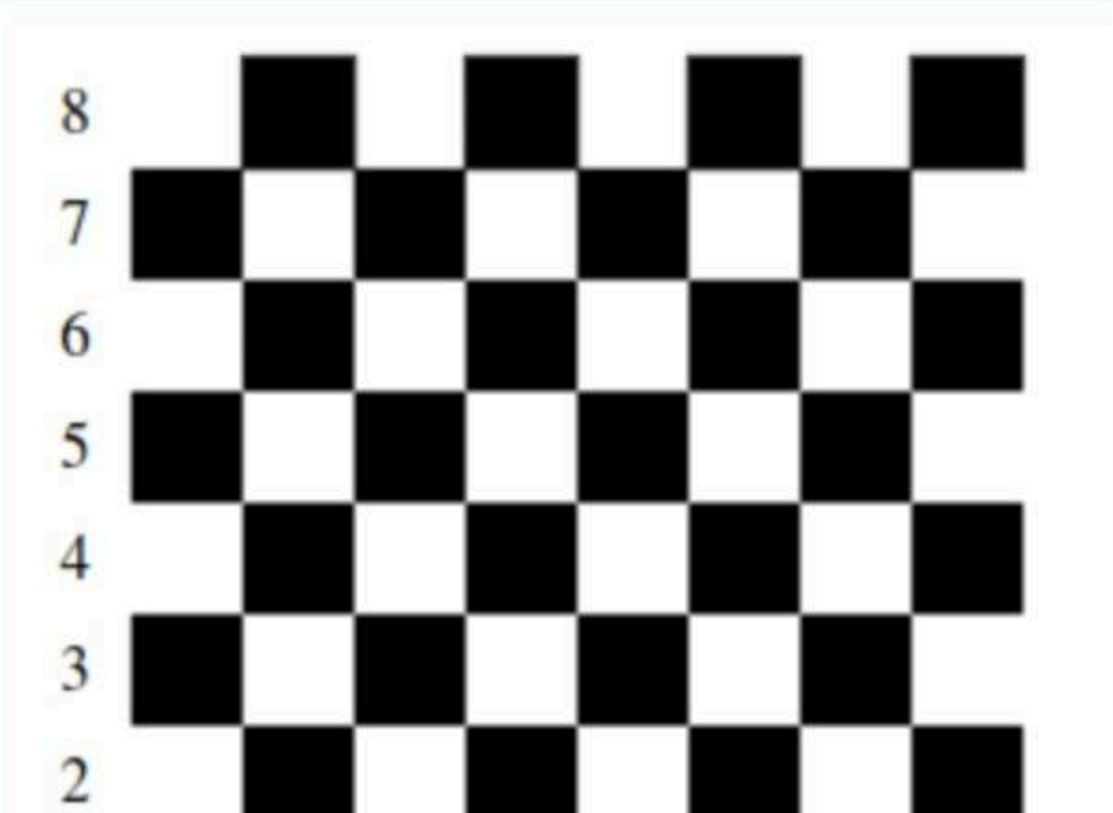
```
{
    printf("Horse");
}
else if(year%12==11)
{
    printf("Sheep");
}
else if(year%12==0)
{
    printf("Monkey");
}
else if(year%12==1)
{
    printf("Rooster");
}
```

```
    }
    else if(year%12==5)
    {
        printf("Ox");
    }
    else if(year%12==6)
    {
        printf("Tiger");
    }
    else
    {
        printf("Hare");
    }
}
```

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

Passed all tests! ✓

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

```c
#include<stdio.h>
int main()
{
    int num,sum;
    char alpha;
    scanf("%c%d",&alpha,&num);
    sum=alpha+num;
    if(sum%2==0)
    {
        printf("The square is black.");
    }
    else
    {
        printf("The square is white.");
    }
```

```
8       if(sum%2==0)
9       {
10          printf("The square is black.");
11      }
12      else
13      {
14          printf("The square is white.");
15      }
16      return 0;
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! ✓

Some data sets specify dates using the year and day of year rather than the year, month, and day of month. The day of year (DOY) is the sequential day number starting with day 1 on January 1st.

There are two calendars – one for normal years with 365 days, and one for leap years with 366 days. Leap years are divisible by 4. Centuries, like 1900, are not leap years unless they are divisible by 400. So, 2000 was a leap year.

To find the day of year number for a standard date, scan down the Jan column to find the day of month, then

```c
#include<stdio.h>
int main()
{
    int d,m,y,feb;
    scanf("%d%d%d",&d,&m,&y);
    if((y%100==0&&y%400)||(y%4==0))
    feb=29;
    else
    feb=28;
    switch(m)
    {
        case 1:
        printf("%d",d);
        break;
        case 2:
```

```
case 2:
printf("%d",31+d);
break;
case 3:
printf("%d",31+feb+d);
break;
case 4:
printf("%d",31+feb+31+d);
break;
case 5:
printf("%d",31+feb+31+30+d);
break;
case 6:
printf("%d",31+feb+31+30+31+d);
break;
```

```
        case 10:
        printf("%d",31+feb+31+30+31+30+31+31+30);
        break;
        case 11:
        printf("%d",31+feb+31+30+31+30+31+31+30+31);
        break;
        case 12:
        printf("%d",31+feb+31+30+31+30+31+31+30+31+30+d);
        break;

    }
```

	Input	Expected	Got	
✓	18 6 2020	170	170	✓

Passed all tests! ✓

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.

```c
#include<stdio.h>
int main()
{
    int a,b;
    char c;
    scanf("%c%d%d",&c,&a,&b);
    switch(c)
    {
        case 'R':
        printf("%d",a*b);
        break;
        case 'S':
        printf("%.0f",(0.5)*a*b);
        break;
        case 'T':
```

```
        printf("%d",a*b);
        break;
        case 'S':
        printf("%.0f",(0.5)*a*b);
        break;
        case 'T':
        printf("%d",a*b);
        break;
        default:
        printf("0");

    }
}
```

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

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: Day Number Name of Day 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: •

Contain a number n (0 < n)

```c
#include<stdio.h>
int main()
{
    int n,day;
    scanf("%d",&n);
    if(n<296)
    day=n;
    else
    day=n-296;
    day%=10;
    day=day+1;
    day%=10;
    switch(day)
    {
```

```
        break;
        case 6:
        printf("Friday");
        break;
        case 7:
        printf("Saturday");
        break;
        case 8:
        printf("Kryptonday");
        break;
        case 9:
        printf("Coluday");
        break;
        case 10:
        printf("Daxamday");
```

```
        case 8:
        printf("Kryptonday");
        break;
        case 9:
        printf("Coluday");
        break;
        case 10:
        printf("Daxamday");
        break;
    }
}
```

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

Passed all tests! ✓