

GE23131-Programming Using C-2024

Attempts allowed: 3

This quiz has been configured so that students may only attempt it using the Safe Exam Browser.

Time limit: 2 hours

Grading method: Highest grade

Your attempts

Attempt 1

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Tuesday, 5 November 2024, 8:38 AM
Duration	48 days 8 hours

Review

The Safe Exam Browser keys could not be validated. Check that you're using Safe Exam Browser with the correct configuration file.

GE23131-Programming Using C-2024

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Tuesday, 5 November 2024, 8:38 AM
Duration	48 days 8 hours

Question 1

Correct

Marked out of 3.00

 [Flag question](#)

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 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     int n1,n2;
5     scanf("%d %d",&n1,&n2);
6     if(n1%10==n2%10)
```

```
1 #include <stdio.h>
2 int main()
3 {
4     int n1,n2;
5     scanf("%d %d",&n1,&n2);
6     if(n1%10==n2%10)
7     {
8         printf("true\n");
9     }
10    else
11    {
12        printf("false\n");
13    }
14    return 0;
15 }
```

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

Passed all tests! ✓

Question 2

Correct

Marked out of 5.00

🚩 Flag question

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*.

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*.

print *Weird*.

Sample Case 1: n = 24

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

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int is_prime(int num)
3 {
4     if(num<=1)
5     {
6         return 0;
7     }
8     for (int i=2;i*i<=num;i++)
9     {
10        if(num%i==0)
11        {
12            return 0;
13        }
14    }
15    return 1;
16 }
17 int main()
18 {
19     int number;
20     scanf("%d",&number);
21     if(number>0&&is_prime(num
22     {
23         printf("Weird");
24     }
25     else
26     {
27         printf("Not Weird");
28     }
29     return 0;
30 }
```

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

Passed all tests! ✓

Question 3

Correct

Marked out of 7.00

 [Flag question](#)

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 $3^2 + 4^2 = 25 = 5^2$. 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>
2 int main()
3 {
4     int a,b,c;
```

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

Passed all tests! ✓

Question 3

Correct

Marked out of 7.00

 [Flag question](#)

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 $3^2 + 4^2 = 25 = 5^2$. 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>
2 int main()
3 {
4     int a,b,c;
5     scanf("%d %d %d",&a,&b,&c);
6     int max,side1,side2;
```

```

7      if(a>=b && b>=c)
8      {
9          max=a;
10         side1=b;
11         side2=c;
12     }
13     else if((b>=a) && (b>=c))
14     {
15         max=b;
16         side1=a;
17         side2=c;
18     }
19     else
20     {
21         max=c;
22         side1=a;
23         side2=b;
24     }
25     if(max*max==side1*side1+side2*side2)
26     {
27         printf("yes\n");
28     }
29     else
30     {
31         printf("no\n");
32     }
33 }
```

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

```
25 }  
26     printf("yes\n");  
27 }  
28 else  
29 {  
30     printf("no\n");  
31 }  
32 return 0;  
33 }
```

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

Passed all tests! ✓

Finish review

GE23131-Programming Using C-2024

Attempts allowed: 2

This quiz has been configured so that students may only attempt it using the Safe Exam Browser.

Time limit: 2 hours

Grading method: Highest grade

Your attempts

Attempt 2

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Friday, 8 November 2024, 2:05 PM
Duration	45 days 3 hours

Review

Attempt 1

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Tuesday, 5 November

GE23131-Programming Using C-2024

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Friday, 8 November 2024, 2:05 PM
Duration	45 days 3 hours

Question 1

Incorrect

Marked out of 3.00

 [Flag question](#)

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

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.

```
1 #include <stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     switch(n)
7     {
8         case 3:
9         {
10             printf("Triangle\n");
11             break;
12         }
13         case 4:
14         {
15             printf("Parallelogram\n");
16             break;
17         }
18         case 5:
19         {
20             printf("Pentagon\n");
21             break;
22         }
23         case 6:
24         {
25             printf("Hexagon\n");
26             break;
27         }
28         case 7:
29         {
30             printf("Heptagon\n");
31             break;
32         }
33         case 8:
34         {
35             printf("Octagon\n");
36             break;
37         }
38         case 9:
39         {
40             printf("Decagon\n");
41             break;
42         }
43     }
44 }
```

```
1 dio.h>
2
3 ▼
4
5 .d",&n);
6 )
7 ▼
8 3:
9 ▼
10 printf("Triangle\n");
11 break;
12
13 4:
14 ▼
15 printf("Parallelograme\n");
16 break;
17
18 5:
19 ▼
20 printf("Pentagon\n");
21 break;
22
23 6:
24 ▼
25 printf("Hexagon\n");
26 break;
27
28 7:
29 ▼
30 printf("Heptagon\n");
31 break;
32
33 8:
34 ▼
35 printf("Octagon\n");
36 break;
37
38 9:
39 ▼
40 printf("Decagon\n");
```

```
28 case 7:  
29 {  
30     printf("Heptagon\n");  
31     break;  
32 }  
33 case 8:  
34 {  
35     printf("Octagon\n");  
36     break;  
37 }  
38 case 9:  
39 {  
40     printf("Decagon\n");  
41     break;  
42 }  
43 default:  
44 {  
45     printf("The number of sides is not valid.\n");  
46     break;  
47 }  
48 }  
49 return 0;  
50 }
```

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

Your code failed one or more hidden tests.

Your code must pass all tests to earn any marks. Try again.

```
28  
29 ▼  
30  
31  
32  
33  
34 ▼  
35  
36  
37  
38  
39 ▼  
40  
41  
42  
43  
44 ▼  
45 sides is not supported.\n");  
46  
47  
48  
49  
50
```

le	✓
on	✓
mber of sides is not supported.	✓

Question 2

Correct

Marked out of 5.00

 [Flag question](#)

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
2002	Horse
2003	Sheep
2004	Monkey
2005	Rooster
2006	Dog
2007	Pig
2008	Rat
2009	Ox
2010	Tiger
2011	Hare

Write a program that reads a year from the

Write a program that reads a year from the user and displays the animal associated with that year. Your program should work correctly for any year greater than or equal to zero, not just the ones listed in the table.

Sample Input 1

2004

Sample Output 1

Monkey

Sample Input 2

2010

Sample Output 2

Tiger

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     int y,dif=0;
5     scanf("%d",&y);
6     dif=(y-2000)%12;
7     switch(dif)
```

```
8     {
9         case 0:
10        {
11            printf("Dragon\n");
12            break;
13        }
14        case 1:
15        {
16            printf("Snake\n");
17            break;
18        }
19        case 2:
20        {
21            printf("Horse\n");
22            break;
23        }
24        case 3:
25        {
26            printf("Sheep\n");
27            break;
28        }
29        case 4:
30        {
31            printf("Monkey\n");
32            break;
33        }
34        case 5:
35        {
36            printf("Rooster\n");
37            break;
38        }
39        case 6:
40        {
41            printf("Dog\n");
42            break;
43        }
44        case 7:
45        {
46            printf("Pig\n");
47            break;
48        }
49        case 8:
50        {
51            printf("Rat\n");
```

```
21     printf("Horse\n");
22     break;
23 }
24 case 3:
25 {
26     printf("Sheep\n");
27     break;
28 }
29 case 4:
30 {
31     printf("Monkey\n");
32     break;
33 }
34 case 5:
35 {
36     printf("Rooster\n");
37     break;
38 }
39 case 6:
40 {
41     printf("Dog\n");
42     break;
43 }
44 case 7:
45 {
46     printf("Pig\n");
47     break;
48 }
49 case 8:
50 {
51     printf("Rat\n");
52     break;
```

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

Passed all tests! ✓

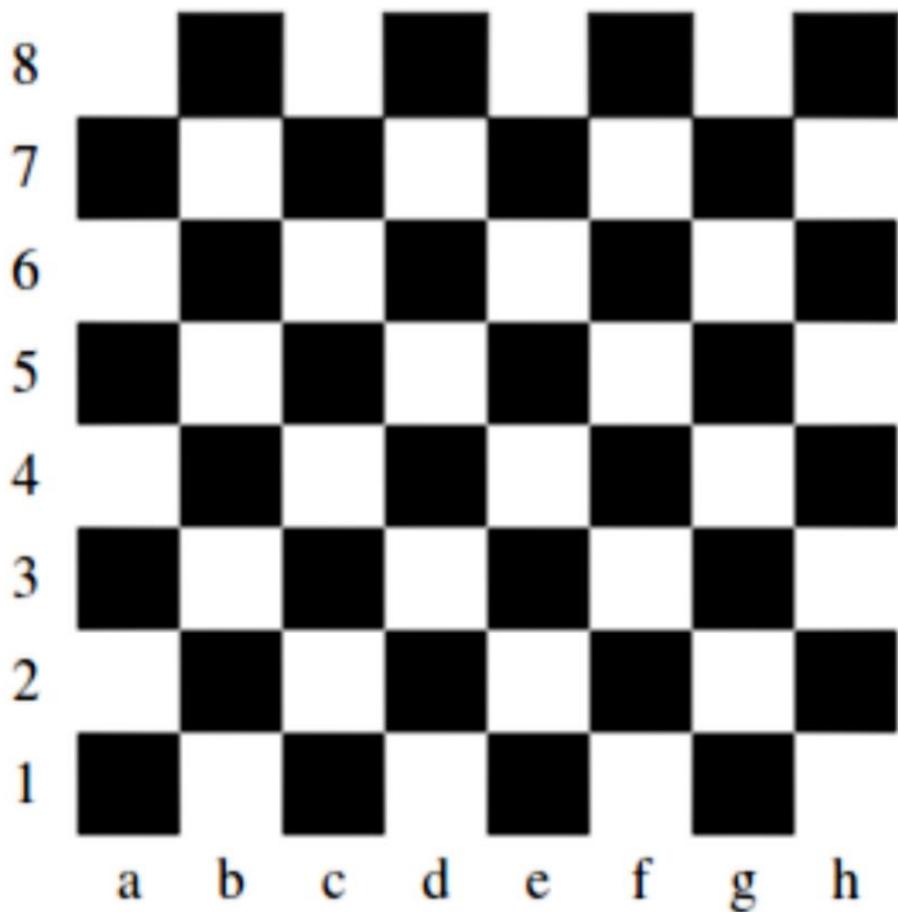
Question 3

Correct

Marked out of 7.00

 [Flag question](#)

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:



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.

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     char ch;
5     int n;
6     scanf("%c %d",&ch,&n);
7     if(ch=='a' || ch=='e' || ch==
8     {
9         if(n%3==0)
10        {
11            printf("The squar
12        }
13    }
14 }
```

Sample Output 2

The square is white.

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     char ch;
5     int n;
6     scanf("%c %d",&ch,&n);
7     if(ch=='a' || ch=='e' || ch=='i')
8     {
9         if(n%3==0)
10        {
11            printf("The square is black");
12        }
13    }
14    else
15    {
16        printf("The square is white");
17    }
18 }
19 else if(ch=='b' || ch=='d' || ch=='o')
20 {
21     if(n%2==0)
22     {
23         printf("The square is black");
24     }
25     else
26     {
27         printf("The square is white");
28     }
29 }
30 }
31
32 return 0;
33 }
```

Sample Output 2

The square is white.

Answer: (penalty regime: 0 %)

```
1 >
2
3 ▼
4
5
6 ,&ch,&n);
7 h=='e'||ch=='g')
8 ▼
9 )
10 ▼
11 f("The square is white.\n");
12
13
14
15 ▼
16 f("The square is black.\n");
17
18
19 b'||ch=='d'||ch=='f'||ch=='h')
20 ▼
21 )
22 ▼
23 f("The square is black.\n");
24
25
26 ▼
27 f("The square is white.\n");
28
29
30
31
32
33
```

```
21 )  
22 ↓  
23 f("The square is black.\n");  
24  
25  
26 ↓  
27 f("The square is white.\n");  
28  
29  
30  
31  
32  
33
```

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

Passed all tests! ✓

Finish review

GE23131-Programming Using C-2024

Attempts allowed: 3

This quiz has been configured so that students may only attempt it using the Safe Exam Browser.

Time limit: 2 hours

Grading method: Highest grade

Your attempts

Attempt 2

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Tuesday, 5 November 2024, 8:51 PM
Duration	47 days 20 hours

Review

Attempt 1

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Tuesday, 5 November

GE23131-Programming Using C-2024

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Tuesday, 5 November 2024, 8:51 PM
Duration	47 days 20 hours

Question 1

Correct

Marked out of 3.00

 [Flag question](#)

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.

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 scan across to the appropriate month column and read the day of year number. Reverse the process to find the standard date for a given day of year.

Write a program to print the Day of Year of a given date, month and year.

Sample Input 1

18

6

2020

Sample Output 1

170

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     int i,inday,inyr,inmon,le
5     scanf("%d %d %d",&inday,&
6         &inmon,&inyr);
```

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 int main()
3 {
4     int i,inday,inyr,inmon,le
5     scanf("%d %d %d",&inday,&
6     char DaysPM[]={31,28,31,3
7     int doy=inday;
8     if (inyr%4==0)
9     {
10         leapyr=1;
11     }
12     else
13     {
14         leapyr=0;
15     }
16     if(inmon>2)
17     {
18         doy=doy+leapyr;
19     }
20     }
21     for(i=0;i<inmon-1;i++)
22     {
23         doy=doy+DaysPM[i];
24     }
25     printf("%d",doy);
26     return 0;
27 }
```

	Input	Expected	Got	
✓	18 6	170	170	✓

Answer: (penalty regime: 0 %)

```
1
2
3 ▼
4 ,inmon,leapyr=0;
5 ,&inday,&inmon,&inyr);
6 1,28,31,30,31,30,31,31,30,31,3
7
8
9 ▼
10
11
12
13 ▼
14
15
16
17 ▼
18 yr;
19
20
21 1;i++)
22 ▼
23 PM[i];
24
25 ;
26
27
```

	Input	Expected	Got	
✓	18	170	170	✓
	6			

Answer: (penalty regime: 0 %)

```
1
2
3 ▾
4 ,leapyr=0;
5 /,&inmon,&inyr);
6 |,30,31,30,31,31,30,31,30,31};
7
8
9 ▾
10
11
12
13 ▾
14
15
16
17 ▾
18
19
20
21
22 ▾
23
24
25
26
27
```

	Input	Expected	Got	
✓	18 6	170	170	✓

	Input	Expected	Got	
✓	18 6 2020	170	170	✓

Passed all tests! ✓

Question 2

Correct

Marked out of 5.00

🚩 Flag question

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

- 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

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

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

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     int l,h,area;
5     char shapex='a';
6     scanf("%c %d %d",&shapex,
7     switch(shapex)
8     {
9         case 'R':
10        {
11            area=l*h;
12            break;
13        }
14        case 'S':
15        {
16            area=(l*h)*0.5;
17            break;
18        }
19        case 'T':
20        {
21            area=l*h;
22            break;
23        }
24        case 'a':
25        {
26            area=0;
27        }
28    }
29    printf("%d",area);
30    return 0;
31 }
32 }
```

	Input	Expected	Got	
✓	T 10	200	200	✓

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int()
3 {
4     float l,h,area;
5     char shapex='a';
6     scanf("%c %f %f",&shapex,&l,&h);
7     switch(shapex)
8     {
9         case 'R':
10        {
11            area=l*h;
12            break;
13        }
14        case 'S':
15        {
16            area=(l*h)*0.5;
17            break;
18        }
19        case 'T':
20        {
21            area=l*h;
22            break;
23        }
24        case 'a':
25        {
26            area=0;
27        }
28
29
30     printf("%d",area);
31     return 0;
32 }
```

	Input	Expected	Got	
✓	T	200	200	✓

	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! ✓

Question 3

Correct

Marked out of 7.00

🚩 [Flag question](#)

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

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 Kryptoday 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 ($0 < n$)

Output format: Print the name of the day you are arriving on

Example Input

7

Example Output

Kryptoday

Example Input

1

Example Output Monday

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     char Day[][12]={"Sunday",
5         int n;
6         scanf("%d",&n);
7         if(n>0){
8             int year=n%296;
9             int mod=year%10;
```

Example Output

Kryptonday

Example Input

1

Example Output Monday

Answer: (penalty regime: 0 %)

1	
2	
3	
4	y" , "Tuesday" , "Wednesday" , "Thur
5	
6	
7	
8	
9	
10	
11	
12	
13	

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

Passed all tests! ✓

Example Output

Kryptonday

Example Input

1

Example Output Monday

Answer: (penalty regime: 0 %)

1	
2	
3	
4	lay", "Kryptonday", "Coluday", "D
5	
6	
7	
8	
9	
10	
11	
12	
13	

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

Passed all tests! ✓

Example Output

Kryptonday

Example Input

1

Example Output Monday

Answer: (penalty regime: 0 %)

```
1
2
3 ▾
4 onday" , "Coluday" , "Daxamday"} ;
5
6
7 ▾
8
9
10
11
12
13
```

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

Passed all tests! ✓

```
1 #include <stdio.h>
2 int main()
3 {
4     char Day[][12]={"Sunday",
5         int n;
6         scanf("%d",&n);
7         if(n>0){
8             int year=n%296;
9             int mod=year%10;
10            printf("%s",Day[mod])
11        }
12    }
13 }
```

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

Passed all tests! ✓

Finish review