Flag question

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main()
{
    printf("Hello C");
    return 0;
}
```

	Expected	Got	
✓	Hello C	Hello C	✓

Passed all tests! ✓

Search
ENG
IN
12:51
14-01-2025

e/mod/quiz/review.php?attempt=317658&cmid=24

Reset answer

```c
#include <stdio.h>

int main()
{
    printf("I love Mangoes");
    return 0;
}
```

	Expected	Got	
✓	I love Mangoes	I love Mangoes	✓

Passed all tests! ✓

Finish review

Search

ENG IN

12:52 14-01-2025

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b;
    float c;
    scanf("%d %d",&a,&b);
    c=((a*12)+b)*2.54;
    printf("%.2f",c);


}
```

	Input	Expected	Got	
✓	5 6	167.64	167.64	✓

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{int a,b,c,d,e,f,h;
scanf("%d%d",&a,&b);
c=a+b;
d=a-b;
h=a*b;
e=a/b;
f=a%b;
printf("%d\n%d\n%d\n%d\n%d\n",c,d,h,e,f);
}
```

	Input	Expected	Got	
✓	100 6	106 94 600 16 4	106 94 600 16 4	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{int a;
float b,c,d;
scanf("%d",&a);
b=a*3.49;
c=0.6*b;
d=b-c;
printf("Regular price: %.2f\n",b);
printf("Discount: %.2f\n",c);
printf("Total: %.2f\n",d);
}
```

	Input	Expected	Got	
✓	10	Regular price: 34.90 Discount: 20.94 Total: 13.96	Regular price: 34.90 Discount: 20.94 Total: 13.96	✓

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int x,y;
    scanf("%d%d",&x,&y);
    if(x<=y)
    {
        printf("YES");
    }
    else
    {
        printf("NO");
    }
    return 0;
}
```

	Input	Expected	Got	
✓	100 110	YES	YES	✓
✓	100 90	NO	NO	✓

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int a,n;
    scanf("%d",&n);
    a=n*(n-1)/2;
    printf("%d",a);
    return 0;
}
```

	Input	Expected	Got	
✓	1	0	0	✓
✓	2	1	1	✓

Passed all tests! ✓

Explanation Out of given numbers, 8 is maximum.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if((a>b)&&(b>c))
    {
        printf("%d",a);
    }
    else if((b>a)&&(b>c))
    {
        printf("%d",b);
    }
    else
    {
        printf("%d",c);
    }
    return 0;
}
```

	Input	Expected	Got	
✓	81 26 15	81	81	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int num1 = 10,num2 = 3;
    printf("Addition Result = %d\n",(num1 + num2));
    printf("Subtraction Result = %d\n",(num1 - num2));
    printf("Multiplication Result = %d\n",(num1 * num2));
    printf("Division Result = %d\n",(num1 / num2));
    printf("Remainder = %d",(num1 % num2));
    return 0;

}
```

	Expected	Got	
✓	Addition Result = 13 Subtraction Result = 7 Multiplication Result = 30 Division Result = 3 Remainder = 1	Addition Result = 13 Subtraction Result = 7 Multiplication Result = 30 Division Result = 3 Remainder = 1	✓

Passed all tests! ✓

```c
#include <stdio.h>

int main()
{
    float num1 = 12.5, num2 = 2.0;
    printf("Result of addition = %f\n", (num1 + num2));
    printf("Result of subtraction = %f\n", (num1 - num2));
    printf("Result of multiplication = %f\n", (num1 * num2));
    printf("Result of division = %f\n", (num1 / num2));
    return 0;
}
```

	Expected	Got	
✓	Result of addition = 14.500000 Result of subtraction = 10.500000 Result of multiplication = 25.000000 Result of division = 6.250000	Result of addition = 14.500000 Result of subtraction = 10.500000 Result of multiplication = 25.000000 Result of division = 6.250000	✓

Passed all tests! ✓

```c
#include <stdio.h>

int main()
{
    char c1 = 'A', c2 = 'D';
    printf("c1 = %d\n", c1);
    printf("c1 + c2 = %d\n", (c1 + c2));
    printf("c1 + c2 + 5 = %d\n", (c1 + c2 + 5));
    printf("Result = %d", (c1 + c2 + '5'));
    return 0;
}
```

	Expected	Got	
✓	c1 = 65 c1 + c2 = 133 c1 + c2 + 5 = 138 Result = 186	c1 = 65 c1 + c2 = 133 c1 + c2 + 5 = 138 Result = 186	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    printf("I love C Language!");
    return 0;
}
```

	Expected	Got	
✓	I love C Language!	I love C Language!	✓

Passed all tests! ✓

The code given below contains text that prints "**DennisRitchieBrianKernighan**".

Answer: (penalty regime: 0 %)

Reset answer

```c
#include <stdio.h>

int main()
{
    printf("Dennis Ritchie \nBrian Kernighan");
    return 0;
}
```

	Expected	Got	
✓	Dennis Ritchie Brian Kernighan	Dennis Ritchie Brian Kernighan	✓

Passed all tests! ✓

Reset answer

```c
#include <stdio.h>

int main()
{
    printf("Orange\n");
    printf("Banana\n");
    //printf("Banana");
    return 0;
}
```

	Expected	Got	
✓	Orange Banana	Orange Banana	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{
    printf("One Two");
    printf("Three\n");
    printf("Four\nFive\n");
    return 0;
}
```

	Expected	Got	
✓	One TwoThree Four Five	One TwoThree Four Five	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int x;
    char y;
    scanf("%d%c",&a&b);
    if(y=='a'||y=='b'||y=='c'||y=='d'||y=='e'||y=='f'||y=='g'||y=='h')
        {
          if(x%2==0)
            {
                printf("The square is white");
            }
            else
            {
                printf("The square is black");
            }
     else if(y=='a'||y=='b'||y=='c'||y=='d'||y=='e'||y=='f'||y=='g'||y=='h')
        {
           if(x%2!=0)
           {
               printf("The square is black");
           }
           else
           {
               printf("The square is white");
           }
        }



}
```

Search
ENG
IN
13:18
14-01-2025

```
            printf("The square is black");
        }
        else
        {
            printf("The square is white");
        }
    }



}
```

Syntax Error(s)

```
__tester__.c: In function ‘main’:
__tester__.c:6:19: error: ‘a’ undeclared (first use in this function)
    6 |     scanf("%d%c",&a&b);
      |                   ^
__tester__.c:6:19: note: each undeclared identifier is reported only once for each function it appears in
__tester__.c:6:21: error: ‘b’ undeclared (first use in this function)
    6 |     scanf("%d%c",&a&b);
      |                     ^
__tester__.c:17:5: error: expected ‘}’ before ‘else’
   17 |     else if(y=='a'||y=='b'||y=='c'||y=='d'||y=='e'||y=='f'||y=='g'||y=='h')
      |     ^~~~
```

Finish review

Reset answer

```c
#include <stdio.h>

int main()
{
    int age = 2;
    int firstNumber = 2;
    int second_number = 3;
    int _i_am_also_a_valid_identifier = 4;
    printf("age = %d\n",age); // Fill in the missing code
    printf("firstNumber = %d\n", firstNumber); // Fill in the missing code
    printf("second_number = %d\n", second_number); // Fill in the missing code
    printf("_i_am_also_a_valid_identifier = %d\n", _i_am_also_a_valid_identifier); // Fill in the missing code
    return 0;
}
```

	Expected	Got	
✓	age = 2 firstNumber = 2 second_number = 3 _i_am_also_a_valid_identifier = 4	age = 2 firstNumber = 2 second_number = 3 _i_am_also_a_valid_identifier = 4	✓

Passed all tests! ✓

Search
ENG
IN

Reset answer

```c
#include <stdio.h>

int main()
{
    printf("Hello, # is a preprocessor in C");
    return 0;
}
```

	Expected	Got	
✓	Hello, # is a preprocessor in C	Hello, # is a preprocessor in C	✓

Passed all tests! ✓

Reset answer

```c
#include <stdio.h>

int main()
{
    printf("Hello, float data type allocates 4 bytes in memory");
    return 0;
}
```

	Expected	Got	
✓	Hello, float data type allocates 4 bytes in memory	Hello, float data type allocates 4 bytes in memory	✓

Passed all tests! ✓

Reset answer

```c
#include <stdio.h>
int main()
{
    printf("Hello, I am learning C Language!");
    return 0;
}
```

	Expected	Got	
✓	Hello, I am learning C Language!	Hello, I am learning C Language!	✓

Passed all tests! ✓

Reset answer

```c
#include<stdio.h>

int main()
{
    printf("Correct Me!");
    return 0;
}
```

	Expected	Got	
✓	Correct Me!	Correct Me!	✓

Passed all tests! ✓

```
    return 0;
}
```

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    printf("Impossible is nothing!");
    return 0;
}
```

	Expected	Got	
✓	Impossible is nothing!	Impossible is nothing!	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int binaryThree = 0b11;
    printf("binaryThree value = %d\n",binaryThree);
    int octalEight = 010;
    printf("octalEight value = %d\n",octalEight);
    int hexTen =0xA;
    printf("hexTen value = %d\n",hexTen);
    int asciiValueOfOne = '1';
    printf("asciiValueOfOne value = %d\n",asciiValueOfOne);
    int asciiValueOfA = 'A';
    printf("asciiValueOfA value = %d\n",asciiValueOfA);
    return 0;

}
```

	Expected	Got	
✓	binaryThree value = 3 octalEight value = 8 hexTen value = 10 asciiValueOfOne value = 49 asciiValueOfA value = 65	binaryThree value = 3 octalEight value = 8 hexTen value = 10 asciiValueOfOne value = 49 asciiValueOfA value = 65	✓

Passed all tests! ✓

Flag question

Reset answer

```c
#include <stdio.h>

int main()
{
    int num1 = 15, num2 = 25, sum;
    printf("Given integers are num1 = %d, num2 = %d\n", num1, num2);
    sum=num1+num2;
    printf("Sum of 2 given numbers = %d\n", sum);
    return 0;
}
```

	Expected	Got	
✓	Given integers are num1 = 15, num2 = 25 Sum of 2 given numbers = 40	Given integers are num1 = 15, num2 = 25 Sum of 2 given numbers = 40	✓

Passed all tests! ✓

Reset answer

```c
#include <stdio.h>

int main()
{
    signed int number1 = -20, number2 = 20;
    unsigned int number3 = -1, number4 = 1;
    printf("Given signed values are %d and %d\n", number1, number2); // Fill the correct format character after %
    printf("Given unsigned values are %u and %d\n", number3, number4); // Fill the correct format character after %
    return 0;
}
```

	Expected	Got	
✓	Given signed values are -20 and 20 Given unsigned values are 4294967295 and 1	Given signed values are -20 and 20 Given unsigned values are 4294967295 and 1	✓

Passed all tests! ✓

Flag question

```c
#include <stdio.h>

int main()
{
    int number1 = 20, number2 = 30,sub;
    sub = number1 - number2;
    printf("The difference of the two given numbers = %d\n", sub);
    return 0;
}

```

	Expected	Got	
✓	The difference of the two given numbers = -10	The difference of the two given numbers = -10	✓

Passed all tests! ✓

```c
#include <stdio.h>

int main()
{
    float num1 = 5.34, num2 = 125.789f, result;
    printf("Given float values are num1 = %f, num2 = %f\n", num1, num2);
    result = num2 / num1;
    printf("The result after dividing in float format = %f\n",result );
    printf("The result after dividing in exponential format = %e\n",result );
    return 0;
}
```

	Expected	Got
✓	Given float values are num1 = 5.340000, num2 = 125.789001 The result after dividing in float format = 23.555992 The result after dividing in exponential format = 2.355599e+01	Given float values are num1 = 5.340000, num2 = 125.789001 The result after dividing in float format = 23.555992 The result after dividing in exponential format = 2.355599

Passed all tests! ✓

Reset answer

```c
#include <stdio.h>

int main()
{
    float num1 = 5.345, num2 = 12.4f, result;
    printf("Given float values are num1 = %f, num2 = %f\n", num1, num2);
    result = num1 / num2;
    printf("Result of division = %f\n", result);
    return 0;
}
```

	Expected	Got	
✓	Given float values are num1 = 5.345000, num2 = 12.400000 Result of division = 0.431048	Given float values are num1 = 5.345000, num2 = 12.400000 Result of division = 0.431048	✓

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int num1 = 7;
    float num2 = 5.5;
    char ch = 'w';
    printf("Result1 = %d\n",(num1 > 5));
    printf("Result2 = %d\n",((num1 + num2) <= 10));
    printf("Result3 = %d\n",(ch == 119));
    printf("Result4 = %d\n",(ch != 'p'));
    printf("Result5 = %d\n",(ch >=10* (num1 + num2)));
    return 0;
}
```

	Expected	Got	
✓	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 1 Result5 = 0	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 1 Result5 = 0	✓

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int num1 = 7;
    float num2 = 5.5;
    char ch = 'w';
    printf("Result1 = %d\n",((num1 >=6)&&(ch == 'w')));
    printf("Result2 = %d\n",((num2 < 11)&&(num1 > 100)));
    printf("Result3 = %d\n",((ch != 'p') || ((num1 + num2) <=10)));
    printf("Result4 = %d\n",!(num1 > (num2 +1)));
    printf("Result5 = %d\n",!(num1 <= 3));
    return 0;

}
```

	Expected	Got	
✓	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 0 Result5 = 1	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 0 Result5 = 1	✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int x = 16;
    printf("+x = %d\n",(+x));
    printf("-x = %d\n",(-x));
    printf("x = %d\n",x);
    printf("++x = %d\n",(++x));
    printf("x = %d\n",x);
    printf("x++ = %d\n",(x++));
    printf("x = %d\n",x);
    printf("--x = %d\n",(--x));
    printf("x = %d\n",x);
    printf("x-- = %d\n",(x--));
    printf("x = %d",x);
    return 0;

}
```

	Expected	Got	
✓	+x = 16 -x = -16 x = 16 ++x = 17 x = 17 x++ = 17 x = 18 --x = 17	+x = 16 -x = -16 x = 16 ++x = 17 x = 17 x++ = 17 x = 18 --x = 17	✓

```c
    printf("-x = %d\n",(-x));
    printf("x = %d\n",x);
    printf("++x = %d\n",(++x));
    printf("x = %d\n",x);
    printf("x++ = %d\n",(x++));
    printf("x = %d\n",x);
    printf("--x = %d\n",(--x));
    printf("x = %d\n",x);
    printf("x-- = %d\n",(x--));
    printf("x = %d",x);
    return 0;

}
```

	Expected	Got	
✓	+x = 16	+x = 16	✓
	-x = -16	-x = -16	
	x = 16	x = 16	
	++x = 17	++x = 17	
	x = 17	x = 17	
	x++ = 17	x++ = 17	
	x = 18	x = 18	
	--x = 17	--x = 17	
	x = 17	x = 17	
	x-- = 17	x-- = 17	
	x = 16	x = 16	

Passed all tests! ✓

Search

```c
#include<stdio.h>
int main()
{
    int x =4,y;
    y = x++;
    printf("y = %d x = %d\n",y,x);
    y= ++x;
    printf("y = %d x = %d\n",y,x);
    y = x--;
    printf("y = %d x = %d\n",y,x);
    y = --x;
    printf("y = %d x = %d\n",y,x);
    return 0;
}
```

	Expected	Got	
✓	y = 4 x = 5 y = 6 x = 6 y = 6 x = 5 y = 4 x = 4	y = 4 x = 5 y = 6 x = 6 y = 6 x = 5 y = 4 x = 4	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int x=24,y=39,z=45;
    z=x+y;
    y=z-y;
    x=z-y;
    printf("x = %d y = %d z = %d",x,y,z);
    return 0;
}
```

	Expected	Got	
✓	x = 39 y = 24 z = 63	x = 39 y = 24 z = 63	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int x =2,y =18,z =12;
    x+= y;
    printf("x = %d\n",x);
    y*= 2;
    printf("y = %d\n",y);
    z/=5;
    printf("z = %d\n",z);
    x %=7;
    printf("x = %d",x);
    return 0;
}
```

	Expected	Got	
✓	x = 20 y = 36 z = 2 x = 6	x = 20 y = 36 z = 2 x = 6	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int marks = 75, pass_marks = 50;
    (marks > pass_marks)? printf("Passed C exam.") : printf("Failed C exam.");
    return 0;

}
```

	Expected	Got	
✓	Passed C exam.	Passed C exam.	✓

Passed all tests! ✓

1.00

Flag question

Reset answer

```c
#include <stdio.h>

int main()
{
    int num1 = 20, num2 = 25, large;
    large = (num1>num2)?num1:num2; // Write the correct code
    printf("Largest number = %d", large);
    return 0;
}
```

	Expected	Got	
✓	Largest number = 25	Largest number = 25	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b;
    scanf("%d %d",&a,&b);
    printf("%d",2*(a+b));
    printf("\n%d",a*b);
    return 0;
}
```

	Input	Expected	Got	
✓	50 20	140 1000	140 1000	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d %d",&a,&b);
    c = a/b;
    printf("%d",c);
    printf("\n%d",a%b);
    return 0;
}
```

	Input	Expected	Got	
✓	60 8	7 4	7 4	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int w,x,y;
    scanf("%d %d %d",&w,&x,&y);
    printf("%d",w*(x-y)-100);
    return 0;
}
```

	Input	Expected	Got	
✓	1000 2 1	900	900	✓

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d",&a);
    b= a/10;
    c= a%10;
    printf("%d",b+c);
    return 0;
}
```

	Input	Expected	Got	
✓	87	15	15	✓
✓	54	9	9	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int binaryThree = 0b11;
    printf("binaryThree value = %d\n" ,binaryThree);
    int octalEight = 010;
    printf("octalEight value = %d\n" ,octalEight);
    int hexTen = 0xA ;
    printf("hexTen value = %d\n",hexTen);
    int asciiValueOfOne = '1';
    printf("asciiValueOfOne value = %d\n",asciiValueOfOne);
    int asciiValueOfA='A';
    printf("asciiValueOfA value = %d\n",asciiValueOfA);
    return 0;
}
```

	Expected	Got	
✓	binaryThree value = 3 octalEight value = 8 hexTen value = 10 asciiValueOfOne value = 49 asciiValueOfA value = 65	binaryThree value = 3 octalEight value = 8 hexTen value = 10 asciiValueOfOne value = 49 asciiValueOfA value = 65	✓

Passed all tests! ✓

```c
#include <stdio.h>

int main()
{
    int num1 = 15, num2 = 25, sum;
    printf("Given integers are num1 = %d, num2 = %d\n", num1, num2);
    sum=num1+num2;
    //Write the code to add num1 and num2 and place the result in the variable sum
    printf("Sum of 2 given numbers = %d\n", sum);
    return 0;
}
```

	Expected	Got	
✓	Given integers are num1 = 15, num2 = 25 Sum of 2 given numbers = 40	Given integers are num1 = 15, num2 = 25 Sum of 2 given numbers = 40	✓

Passed all tests! ✓

Finish review

Flag question

Answer: (penalty regime: 0 %)

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

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

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main(){
    int n;
    scanf("%d",&n);
    if(1<=n && n<=100){
        if(n%2==0){
            printf("Not Weird");
            if(6<=n && n<=20){
                printf("Weird");
            }
        }
        else{
            printf("Weird");
        }
    }
}
```

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

```c
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if((a*a+b*b==c*c)||(a*a+c*c==b*b)||(b*b+c*c==a*a))
    {
        printf("yes");
    }
    else
    {
        printf("no");
    }
}
```

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

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int shape;
    scanf("%d",&shape);
    switch(shape)
    {
        case 3:
            {
                printf("Triangle");
                break;
            }
        case 4:
            {
                printf("Square");
                break;
            }
        case 5:
        {
            printf("Pentagon");
            break;
        }
        case 6:
        {
            printf("Hexagon");
            break;
        }
        case 7:
        {
            printf("Heptagon");
            break;
        }
        case 8:
        {
            printf("Octagon");
            break;
        }
        case 9:
```

```
        break;
    }
    case 9:
    {
        printf("Nonagon");
        break;
    }
    case 10:
    {
        printf("Decagon");
        break;
    }
    default:
    {
        printf("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.	✓

Passed all tests! ✓

Question **2**

Correct

The Chinese zodiac assigns animals to years in a 12-year cycle. One 12-year cycle is shown in the table below. The pattern

```c
#include<stdio.h>
int main()
{
    int year;
    scanf("%d",&year);
    switch(year%12)
    {

        case 8:
        {
            printf("Dragon");
            break;
        }
        case 9:
        {
            printf("Snake");
            break;
        }
        case 10:
        {
            printf("Horse");
            break;
        }
        case 11:
        {
            printf("Sheep");
            break;
        }
        case 0:
        {
            printf("Monkey");
            break;
        }
        case 1:
        {
            printf("Roaster");
            break;
        }
```

```c
            break;
        }
        case 1:
        {
            printf("Roaster");
            break;
        }
        case 2:
        {
            printf("Dog");
            break;
        }
        case 3:
        {
            printf("Pig");
            break;
        }
        case 4:
        {
            printf("Rat");
            break;
```

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

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int d,month,year,feb;
    scanf("%d%d%d",&d,&month,&year);
    if((year%4==0 && year%100!=0) || (year%400==0))
    {
        feb=29;
    }
    else
    {
        feb=28;
    }
    switch(month)
    {
        case 1:
        printf("%d",d);
        break;
        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 7:
        printf("%d",31+feb+31+30+31+30+d);
        break;
```

```
        break;
        case 7:
        printf("%d",31+feb+31+30+31+30+d);
        break;
        case 8:
        printf("%d",31+feb+31+30+31+30+31+d);
        break;
        case 9:
        printf("%d",31+feb+31+30+31+30+31+31+d);
        break;
        case 10:
        printf("%d",31+feb+31+30+31+30+31+31+30+d);
        break;
        case 11:
        printf("%d",31+feb+31+30+31+30+31+31+30+31+d);
        break;
        case 12:
        printf("%d",31+feb+31+30+31+30+31+31+30+31+30+d);
        break;
        default :
```

	Input	Expected	Got	
✓	18 6 2020	170	170	✓

Passed all tests! ✓

(penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    char c;
    int a,b;
    scanf("%c%d%d",&c,&a,&b);
    // printf("%d%d",a,b);
    if(c=='R')
    {
        printf("%d",(a*b));
    }
    else if(c=='S')
    {
        printf("%d",((a*b)/2));
    }
    else if(c=='T')
    {
        printf("%d",a*b);
        }
     else
     {
         printf("0");
     }
}
```

	Input	Expected	Got	
✓	T	200	200	✓

```
        printf("0");
    }
}
```

	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! ✓

```c
#include<stdio.h>
int main()
{
    int x,a,y;
    scanf("%d",&x);
    y=x%296;
    a=y%10;
    a=a+1;
    switch(a)
    {
        case 1:
        printf("Sunday");
        break;
        case 2:
        printf("Monday");
        break;
        case 3:
        printf("Tuesday");
        case 4:
        printf("Wednesday");
        break;
        case 5:
        printf("Thursday");
        break;
        case 6:
        printf("Friday");
        case 7:
        printf("Saturday");
        break;
        case 8:
        printf("Kryptonday");
        break;
        case 9:
        printf("Clouday");
        break;
        case 10:
        printf("Daxamday");
        break;
```

```
        case 9:
        printf("Clouday");
        break;
        case 10:
        printf("Daxamday");
        break;

    }


}
```

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

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
int n,i;
int largest,second_largest;
scanf("%d",&n);
int arr[n];
for(i=0;i<n;i++)
{
scanf("%d", &arr[i]);
}
if (arr[0] >arr[1])
{
largest = arr[0];
second_largest = arr[1];
}
else
{
largest = arr[1];
second_largest = arr[0];
}
for(i=2;i<n;i++)
{
if(arr[i]>largest)
{
second_largest =largest;
largest = arr[i];
}
else if(arr[i] >second_largest && arr[i] != largest)
{
second_largest = arr[i];
}
}
printf("The largest element of the array = %d\n",largest);
printf("The second largest element of the array = %d\n", second_largest);

return 0;
}
```

	Input	Expected	Got	
✓	5 2 5 9 7 3	The largest element of the array = 9 The second largest element of the array = 7	The largest element of the array = 9 The second largest element of the array = 7	✓
✓	6 64 87 34 58 62 18	The largest element of the array = 87 The second largest element of the array = 64	The largest element of the array = 87 The second largest element of the array = 64	✓

Passed all tests! ✓

Write a program to find the minimum and second minimum elements with in the elements of one dimensional array.

Constraints:

- $1 <= N <= 10^3$
- $1 <=$ Elements of the array $<= 10^6$

```c
#include<stdio.h>
int main()
{
    int N,i;
    int min,second_min;
    scanf("%d",&N);
    int arr[N];
    for(i=0;i<N;i++)
    {
        scanf("%d",&arr[i]);
    }
    min = arr[0];
    second_min=arr[0];
    for(i=1;i<N;i++)
    {
        if(arr[i] < min)
        {
            second_min = min;
            min = arr[i];
        }
        else if(arr[i] < second_min && arr[i] != min)
        {
            second_min = arr[i];

        }
    }
    printf("Min element = %d\n", min);
    printf("Second min element = %d\n", second_min);


}
```

```
    printf("Min element = %d\n", min);
    printf("Second min element = %d\n", second_min);


}
```

	Input	Expected	Got	
✓	4 65 32 85 96	Min element = 32 Second min element = 65	Min element = 32 Second min element = 65	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 1.00

Flag question

Write a program to read a student n subjects marks in an array and find the total, average of the marks.

For example, if the user gives the **input** as:

3

Next, the program should print the messages one by one on the console.
if the user gives the **input** as:

```
The average marks = 43.750000
45
38
56
```

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,i,total =0;
    float average;

    scanf("%d", &n);
    int marks[n];
    for(i=0;i<n;i++)
    {
        scanf("%d", &marks[i]);
        total += marks[i];
    }
    average = (float)total /n;

    printf("The total marks = %d\n",total);
    printf("The average marks = %f\n", average);

    return 0;

}
```

```
    average = (float)total /n;

    printf("The total marks = %d\n",total);
    printf("The average marks = %f\n", average);

    return 0;

}
```

	Input	Expected	Got	
✓	5 45 65 55 75 85	The total marks = 325 The average marks = 65.000000	The total marks = 325 The average marks = 65.000000	✓
✓	4 36 45 38 56	The total marks = 175 The average marks = 43.750000	The total marks = 175 The average marks = 43.750000	✓

Passed all tests! ✓

```c
#include <stdio.h>
int main()
{
    int i, j, m, n, p, q;
    int a[5][5], b[5][5], c[5][5];
    scanf("%d %d", &m, &n);
    for (i=0;i<m;i++ )
    {
        for (j=0;j<n;j++ )
        {
            scanf("%d", &a[i][j]);
        }
    }
    scanf("%d %d", &p, &q);
    for (i=0;i<p;i++ )
    {
        for (j=0;j<q;j++)
        {
            scanf("%d", &b[i][j]);
        }
    }
    printf("The given matrix-1 is\n");
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        {
            printf("%d ",a[i][j]);
        }
        printf("\n");
    }
    printf("The given matrix-2 is\n");
    for(i=0;i<p;i++)
    {
        for(j=0;j<q;j++)
        {
            printf("%d ",b[i][j]);
        }
```

```c
    {
        for(j=0;j<q;j++)
        {
            printf("%d ",b[i][j]);
        }
        printf("\n");
    }

    if (m==p && n==q )
    {
        for (i=0;i<m;i++)
        {
            for (j=0;j<n;j++)
            {
                c[i][j] =a[i][j]+b[i][j] ;
            }
        }
        printf("Addition of two matrices is\n");
        for(i=0;i<m;i++)
        {
```

	Input	Expected	Got	
✓	2 2 1 2 3 4 2 2 4 5 6 7	The given matrix-1 is 1 2 3 4 The given matrix-2 is 4 5 6 7 Addition of two matrices is 5 7 9 11	The given matrix-1 is 1 2 3 4 The given matrix-2 is 4 5 6 7 Addition of two matrices is 5 7 9 11	✓

Passed all tests! ✓

```c
#include <stdio.h>
int main()
{
    int i, j, k, m, n, p, q;
    int a[5][5], b[5][5], c[5][5];
    scanf("%d %d", &m, &n);
    for (i=0;i<m;i++)
    { // Complete the code in for
        for (j=0;j<n;j++)
        { // Complete the code in for
            scanf("%d", &a[i][j]);
        }
    }
    scanf("%d %d", &p, &q);
    for (i=0;i<p;i++)
    { // Complete the code in for
        for (j=0;j<q;j++)
        { // Complete the code in for
            scanf("%d", &b[i][j]);
        }
    }
    printf("The given matrix-1 is\n");
    // Write the code to display Matrix-1 elements
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        {
            printf("%d ",a[i][j]);
        }
        printf("\n");
    }


    printf("The given matrix-2 is\n");
    // Write the code to display Matrix-2 elements
    for(i=0;i<p;i++)
    {
        for(j=0;j<q;j++)
```

```c
    printf("The given matrix-2 is\n");
    // Write the code to display Matrix-2 elements
    for(i=0;i<p;i++)
    {
        for(j=0;j<q;j++)
        {
            printf("%d ",b[i][j]);
        }
        printf("\n");
    }


    if (n==p )
    { // Write the condition part
        for ( i=0;i<m;i++)
        { // Complete the code in for
            for (j=0;j<q;j++)
            { // Complete the code in for
                c[i][j] = 0; // Complete the statement
```

	Input	Expected	Got	
✓	2 3 1 2 3 5 7 9 3 2 3 2 1 5 6 7	The given matrix-1 is 1 2 3 5 7 9 The given matrix-2 is 3 2 1 5 6 7 Multiplication of two matrices is 23 33 76 108	The given matrix-1 is 1 2 3 5 7 9 The given matrix-2 is 3 2 1 5 6 7 Multiplication of two matrices is 23 33 76 108	✓

Passed all tests! ✓

with Strings: Attempt review | REC-CIS - Personal - Microsoft Edge

www.rajalakshmicolleges.org/moodle/mod/quiz/review.php?attempt=159349&cmid=79

```c
#include <stdio.h>
#include<ctype.h>

int main()
{
    char line[100];
    int i, vowels = 0, consonants = 0, digits = 0, spaces = 0;
    fgets(line, sizeof(line), stdin);
    for (i=0 ;line[i] !='\0' ;i++ )
    { // Complete the code in for
        if (line[i] == 'a' || line[i] == 'e' || line[i] == 'i' ||
            line[i] == 'o' || line[i] == 'u' || line[i] == 'A' ||
            line[i] == 'E' || line[i] == 'I' || line[i] == 'O'||
            line[i] == 'U')
        { // Write the condition part
            ++vowels;
        }
        else if ((line[i]>='a' && line[i]<='z') || (line[i]>='A' && line[i]<='Z'))
        { // Write the condition part
            ++consonants;
        }
        else if (line[i]>='0' && line[i]<='9')
        { // Write the condition part
            ++digits;
        }
        else if (line[i] == ' ' )
        { // Write the condition part
            ++spaces;
        }
    }
    printf("Vowels = %d\n", vowels);
    printf("Consonants = %d\n",consonants);
    printf("Digits = %d\n",digits);
    printf("White spaces = %d\n", spaces);
    return 0;
}
```

Search

ENG IN

14:30 14-01-2025

al - Microsoft Edge

/mod/quiz/review.php?attempt=159349&cmid=79

	Input	Expected	Got	
✓	kohli hits 100 in every cricket match!	Vowels = 9 Consonants = 19 Digits = 3 White spaces = 6	Vowels = 9 Consonants = 19 Digits = 3 White spaces = 6	✓

Passed all tests! ✓

on **2**

t

d out of

question

Fill in the missing code in the below sample code which copies a given string into another string.

Initially, read a string from the standard input device and write a loop to copy each character of given string into another st the string is reached.

Place '\0' at the end of the copied string.

Finally, the copied string is displayed on the screen.

For example:

```c

int main()
{
    int i,j;
    char a[20],b[20],c[40];
    scanf("%s",a);
    scanf("%s",b);

    for(i=0;a[i] !='\0';i++)
    {
        c[i]=a[i];
    }
    for(j=0;b[j]!='\0';j++)
    {
        c[i + j]=b[j];
    }
    c[i + j] ='\0';
    printf("The copied string = %s\n", c);
    return 0;

}
```

	Input	Expected	Got	
✓	GangaRiver	The copied string = GangaRiver	The copied string = GangaRiver	✓

Passed all tests! ✓

```c
#include <stdio.h>

int main()
{
    char a[20], b[20], c[40];
    int i, j;
    scanf("%s", a);
    scanf("%s", b);
    for (i=0;a[i]!='\0';i++)
    { // Complete the code in for
        c[i] = a[i]; //Complete the statement
    }
    for (j=0;b[j]!='\0';j++ )
    { // Complete the code in for
        c[i + j] =b[j] ; //Complete the statement

    }
    c[i + j] = '\0' ; //Complete the statement
    printf("%s\n", c);
    return 0;
}
```

	Input	Expected	Got	
✓	Narendra Modi	NarendraModi	NarendraModi	✓

Passed all tests! ✓

Reset answer

```c
#include <stdio.h>

int main()
{
    char a[20], b[20];
    int i = 0, flag = 0;
    scanf("%s", a);
    scanf("%s", b);
    while (a[i] !='\0' && b[i]!= '\0')
    { //Complete the condition part
        if (a[i] !=b[i] )
        { //Complete the condition part
            flag = 1; //Complete the statement
            break;
        }
        i++;
    }
    if (flag == 0 && a[i] =='\0' && b[i] == '\0')
    { //Complete the condition part
        printf("Two strings are equal\n");
    }
    else
    {
        printf("Two strings are not equal\n");
    }
    return 0;
}
```

	Input	Expected	Got	
✓	Godavari Godavari	Two strings are equal	Two strings are equal	✓
✓	Narmada narmada	Two strings are not equal	Two strings are not equal	✓

Passed all tests! ✓

Question **5**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample code to search the occurrence of a given character in

Read a string and a character from the standard input device and write a loop to check each char

If the given character is equal to a character in the string then increment the count with in the loo

Finally, display the count variable which has the total number of occurrences of the given charact

For example:

```c
#include <stdio.h>

int main()
{
    char str[20], ch;
    int count = 0, i;
    scanf("%s", str);
    scanf(" %c", &ch);
    for (i=0;str[i]!='\0';i++ )
    { // Complete the code in for
        if (str[i]==ch )
        { // Write the condition part
            count++;
        }
    }
    if (count==0 )
    { // Write the condition part
        printf("The character '%c' is not presented in the string %s\n", ch, str);
    }
    else
    {
        printf("Occurence of character '%c' in the given string %s = %d\n", ch, str, count);
    }
    return 0;
}
```

	Input	Expected	Got
✓	CurrencyDemonitisation n	Occurence of character 'n' in the given string CurrencyDemonitisation = 3	Occurence of character

Passed all tests! ✓

Search
ENG
IN

```c
#include<stdio.h>

int main()
{
    int upper_count = 0, lower_count = 0;
    char ch[80];
    int i;
    scanf("%s",ch); // Complete the statement
    i =0 ; // Complete the statement
    while (ch[i] !='\0' )
    { // Write the condition part
        if (ch[i]>='A' && ch[i]<='Z')
        { // Write the condition part
            upper_count++;
        }
        if (ch[i]>='a' && ch[i] <='z')
        { // Write the condition part
            lower_count++;
        }
        i++;
    }
    printf("Number of uppercase letters = %d\n",upper_count);
    printf("Number of lowercase Letters = %d\n",lower_count);
    return 0;
}
```

	Input	Expected	Got	
✓	KrishnaAndGodavariAreRivers	Number of uppercase letters = 5 Number of lowercase Letters = 22	Number of uppercase letters = 5 Number of lowercase Letters = 22	✓

Passed all tests! ✓

Reset answer

```c
#include<stdio.h>

int main()
{
    char ch[80], temp;
    int i, j;
    scanf("%s", ch);
    i = j = 0;
    while (ch[j]!='\0')
    { // Write the condition part
        j++;
    }
    j--;
    while (i<j )
    { // Write the condition part
        temp =ch[i] ; // Complete the statement
        ch[i] =ch[j]; // Complete the statement
        ch[j] = temp; // Complete the statement
        i++;
        j--;
    }
    printf("The reverse of a given string : %s\n", ch);
    return 0;
}
```

	Input	Expected	Got	
✓	Software	The reverse of a given string : erawtfoS	The reverse of a given string : erawtfoS	✓

Passed all tests! ✓

```c
#include <stdio.h>
#include <string.h>

int main()
{
    char ch[80];
    int i, j, length, flag = 0;
    scanf("%s",ch); // Complete the statement
    length = strlen(ch);
    for (i=0,j=length-1;i<j;i++,j-- )
    { // Complete the code in for
        if (ch[i]!= ch[j])
        { // Write the condition part
            flag=1;
            break;
        }
    }
    if (flag ==0 )
    { // Write the condition part
        printf("The given string %s is a palindrome\n",ch  ); // Complete the statement
    }
    else
    {
        printf("The given string %s is not a palindrome\n",ch  ); // Complete the statement
    }
    return 0;
}
```

Input	Expected	Got

```
    { // Complete the code in for
        if (ch[i]!= ch[j])
        { // Write the condition part
            flag=1;
            break;
        }
    }
    if (flag ==0 )
    { // Write the condition part
        printf("The given string %s is a palindrome\n",ch  ); // Complete the statement
    }
    else
    {
        printf("The given string %s is not a palindrome\n",ch  ); // Complete the statement
    }
    return 0;
}
```

	Input	Expected	Got	
✓	12321	The given string 12321 is a palindrome	The given string 12321 is a palindrome	✓
✓	amaravathi	The given string amaravathi is not a palindrome	The given string amaravathi is not a palindrome	✓

Passed all tests! ✓

Reset answer

```c
#include <stdio.h>
#include <string.h>

int main()
{
    char ch[20];
    scanf("%s", ch);
    printf("The length of the string %s is %ld\n", ch, strlen(ch)); //Correct the code
    return 0;
}
```

	Input	Expected	Got	
✓	NarendraModi	The length of the string NarendraModi is 12	The length of the string NarendraModi is 12	✓

Passed all tests! ✓

Question **2**

The function strcpy() is used to **copy** one string into another string including the NULL character (terminator char '\0')

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<string.h>
int main()
{
    char str1[20], str2[20];
    scanf("%s",str2);
    strcpy(str1,str2);
    printf("The copied string = %s\n", str1);
    return 0;
}
```

	Input	Expected	Got	
✓	Rose	The copied string = Rose	The copied string = Rose	✓

Passed all tests! ✓

Question **3**

The function strcat() is used to concatenate two strings into a single string

Search

```c
#include <stdio.h>
#include <string.h>

int main()
{
    char a[20], b[20];
    scanf("%s", a);
    scanf("%s", b);
    if(strcmp(a,b) == 0)
    {
        printf("The given two strings are equals.\n");
    }
    else if (strcmp(a, b) > 0)
    {
        printf("%s%s\n",a, b);

    }
    else
    {
        printf("%s %s.\n",b,a);
    }
    return 0;
}
```

	Input	Expected	Got	
✓	REC Chennai	RECChennai	RECChennai	✓

Passed all tests! ✓

Search

ENG
IN

REC

Answer: (penalty regime: 0 %)

Reset answer

```c
#include <stdio.h>
#include <string.h>

int main()
{
    char a[20], b[20];

    scanf("%s", a);
    scanf("%s", b);
    //Compare two strings

    if (strcmp(a,b) == 0)
    { // Correct the code
        printf("The given two strings are equal\n");
    }
    else if (strcmp(a, b) > 0 )
    { // Correct the code
        printf("The string %s is higher than the string %s\n", a, b);
    }
    else
    {
        printf("The string %s is higher than the string %s\n", b, a);
    }
    return 0;
}
```

Input	Expected	Got

```c
    else if (strcmp(a, b) > 0 )
    { // Correct the code
        printf("The string %s is higher than the string %s\n", a, b);
    }
    else
    {
        printf("The string %s is higher than the string %s\n", b, a);
    }
    return 0;
}
```

	Input	Expected	Got
✓	NarendraModi narendramodi	The string narendramodi is higher than the string NarendraModi	The string narendramodi is higher than
✓	Krishna Godavari	The string Krishna is higher than the string Godavari	The string Krishna is higher than the s
✓	REC REC	The given two strings are equal	The given two strings are equal

Passed all tests! ✓

Fini

Search

ENG
IN

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
int t,n,a;
scanf("%d",&t);
while(t--)
{
scanf("%d",&n);
a=n/4;
if((n%2==0&&a%2!=0)||n==1||n==3)
{
printf("Yes\n");
}
else
{
printf("No\n");
}
}
return 0;
}
```

	Input	Expected	Got	
✓	3 1 6 7	Yes Yes No	Yes Yes No	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int c(int digit)
{
    int h[10]={1,0,0,0,1,0,1,0,2,1};
    return h[digit];
}
int main()
{
    int num;
    scanf("%d",&num);
    int t=0;
    while(num>0)
    {
        int digit=num%10;
        t+=c(digit);
        num/=10;
    }
    printf("%d",t);
    return 0;
}
```

	Input	Expected	Got	
✓	630	2	2	✓
✓	1288	4	4	✓

```c
#include<stdio.h>
int main()
{
    int n,s=0,c=1,co=0;
    scanf("%d",&n);
    do
    {
        s+=c;
        c*=2;
        co++;

    }while(s<n);
    printf("%d",co);
}
```

	Input	Expected	Got	
✓	10	4	4	✓
✓	5	3	3	✓
✓	20	5	5	✓
✓	500	9	9	✓
✓	1000	10	10	✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{   int i,n=0;
    while(scanf("%d",&i)==1)
    {
        if(i%2!=0)
        {
            n++;
        }
    }
    printf("%d",n);
}
```

	Input	Expected	Got	
✓	5 10 15 20 25 30 35 40 45 50	5	5	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b;
    scanf("%d",&a);
    b=a%10;
    while(b>0)
     if(b==0||b==1||b==6||b==8||b==9)
     {
         printf("true");
         break;
     }
     else
     {
         printf("false");
         break;
     }
}
```

	Input	Expected	Got	
✓	6	true	true	✓
✓	89	true	true	✓
✓	25	false	false	✓

Passed all tests! ✓

Explanation 2

$2 + 3 = 5$, *is* the best case for maximum nutrients.

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{
    long long int x,y,sum=0,i;
     scanf("%lld %lld",&x,&y);
     for(i=1;i<=x;i++)
     {
         sum+=i;
         if(sum==y)
         {
             sum=sum-1;

         }
     }
     printf("%lld",sum%1000000007);
}
```

```
}
```

	Input	Expected	Got	
✓	2 2	3	3	✓
✓	2 1	2	2	✓
✓	3 3	5	5	✓

Passed all tests! ✓

Search

```c
#include<stdio.h>
int main()
{
    int t,c,z=0;
    scanf("%d",&t);
    while(z<t)
    {
    scanf("%d",&c);
        for(int i=1;i<=c;i++)
        {
            for(int j=1;j<=c;j++)
            {
                if((i%2==0 && j%2==0) || (i%2!=0 && j%2!=0))
                {
                    printf("W");

                }
                else
                {
                    printf("B");
                }
            }
            printf("\n");
        }


    z++;
    if(z>t)
    {
        break;
    }
    }
}
```

	Input	Expected	Got	
✓	2	WBW	WBW	✓
	3	BWB	BWB	
	5	WBW	WBW	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	
		BWBWB	BWBWB	
		WBWBW	WBWBW	

Passed all tests! ✓

Question **2**

Correct

Marked out of

.00

Let's print a chessboard!

Write a program that takes input:

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int c,t,z=0;
    char x;
    scanf("%d",&t);
    while(z<t)
    {
        scanf("%d %c",&c,&x);
        for(int i=1;i<=c;i++)
        {
            for(int j=1;j<=c;j++)
            {
                if(x=='W')
                {
                    if((i%2==0 && j%2==0) || (i%2!=0  && j%2!=0))
                    {
                        printf("W");
                    }
                    else
                    {
                        printf("B");
                    }
                }
                    else if(x=='B')
                    {
                        if((i%2==0 && j%2==0) || (i%2!=0 && j%2!=0))
                        {
                            printf("B");
                        }
                        else
                        {
```

```
                    {
                        printf("W");
                    }
                    else
                    {
                        printf("B");
                    }
                }
                    else if(x=='B')
                    {
                        if((i%2==0 && j%2==0) || (i%2!=0 && j%2!=0))
                        {
                            printf("B");
                        }
                        else
                        {
                            printf("W");
                        }
                    }
            }
            printf("\n");
        }
    t--;
    if(z==t)
    {
        break;
    }
    }
}
```

	Input	Expected	Got	
✓	2 2 W 3 B	WB BW BWB WBW BWB	WB BW BWB WBW BWB	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 7.00

Flag question

Decode the logic and print the Pattern that corresponds to given input.

If N= 3

then pattern will be :

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,v,p3,c,i,in,i1,i2,t,ti;
    scanf("%d",&t);
    for(ti=0;ti<t;ti++)
    {
        v=0;
        scanf("%d",&n);
        printf("Case #%d\n",ti+1);
        for(i=0;i<n;i++)
        {
                c=0;
                if(i>0)
                {
                    for(i1=0;i1<i;i1++)
                    printf("**");
                }
                for(i1=i;i1<n;i1++)
                {
                    if(i>0)c++;
                    printf("%d0",++v);
                }
                if(i==0)
                {
                    p3=v+(v*(v-1))+1;
                    in=p3;
                }
                in=in-c;
                p3=in;
                for(i2=i;i2<n;i2++)
                {
                    printf("%d",p3++);
```

```
                {
                    for(i1=0;i1<i;i1++)
                    printf("**");
                }
                for(i1=i;i1<n;i1++)
                {
                    if(i>0)c++;
                    printf("%d0",++v);
                }
                if(i==0)
                {
                    p3=v+(v*(v-1))+1;
                    in=p3;
                }
                in=in-c;
                p3=in;
                for(i2=i;i2<n;i2++)
                {
                    printf("%d",p3++);
                    if(i2!=n-1)
                    printf("0");
                }
                printf("\n");

            }
        }
}
```

	Input	Expected	Got	
✓	3 3 4 5	Case #1 10203010011012 **4050809 ****607 Case #2 1020304017018019020 **50607014015016 ****809012013 ******10011 Case #3 102030405026027028029030 **6070809022023024025 ****10011012019020021 ******13014017018 ********15016	Case #1 10203010011012 **4050809 ****607 Case #2 1020304017018019020 **50607014015016 ****809012013 ******10011 Case #3 102030405026027028029030 **6070809022023024025 ****10011012019020021 ******13014017018 ********15016	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b=0,c,e=0;
    scanf("%d",&a);
    if(a<0)
    a=-a;
    c=a;
    while(c!=0)
    {
        c/=10;
        e++;

    }
    c=a;
    while(c!=0)
    {
        int d=c%10;
        int f=1;
        for(int i=0;i<e;i++)
        {
            f*=d;
        }
        b+=f;
        c/=10;
    }
    if(a==b)
    {
        printf("true");
    }
    else
    {
        printf("false");
    }
}
```

```
    if(a==b)
    {
        printf("true");
    }
    else
    {
        printf("false");
    }
}
```

	Input	Expected	Got	
✓	153	true	true	✓
✓	123	false	false	✓

Passed all tests! ✓

Question **2**

Correct

Marked out of

Take a number, reverse it and add it to the original number until the obtaine

Sample Input 1 32 Sample Output 1 55 Sample Input 2 789 Sample Output

```c
#include<stdio.h>
int main()
{
    int rn,n,nt=0,i=0;
    scanf("%d",&n);
    do
    {
        nt=n;
        rn=0;
        while(n!=0)
        {
            rn =rn*10+n%10;
            n=n/10;
        }
        n=nt+rn;
        i++;
    }
    while(rn!=nt || i==1);

    printf("%d",rn);
    return 0;
}
```

	Input	Expected	Got	
✓	32	55	55	✓
✓	789	66066	66066	✓

```c
#include<stdio.h>
int con(int a)
{
    int c=a;
    while(c!=0)
    {
        int d=c%10;
        if(d!=3 && d!=4)
        return 0;
        c/=10;
    }
    return 1;
}
int main()
{
    int a,b=0;
    scanf("%d",&a);
    while(a!=0)
    {
        b++;
        if(con(b))
        {
            a--;
        }
    }
    printf("%d",b);
    return 0;
}
```

	Input	Expected	Got	
✓	34	33344	33344	✓

Passed all tests! ✓

REC-CIS

```
#include<stdio.h>
int main()
{
    int a;
    scanf("%d",&a);
    for(int i=1; i<11; i++)
    {
        printf("%d x %d = %d\n", a, i, (a*i));
    }
}
```

	Input	Expected	Got	
✓	2	2 x 1 = 2 2 x 2 = 4 2 x 3 = 6 2 x 4 = 8 2 x 5 = 10 2 x 6 = 12 2 x 7 = 14 2 x 8 = 16 2 x 9 = 18 2 x 10 = 20	2 x 1 = 2 2 x 2 = 4 2 x 3 = 6 2 x 4 = 8 2 x 5 = 10 2 x 6 = 12 2 x 7 = 14 2 x 8 = 16 2 x 9 = 18 2 x 10 = 20	✓

Falling back to raw text area.

```
#include<stdio.h>
int main()
{
    long int a,b,c=0;
    scanf("%ld", &a);
    scanf("%ld", &b);

    for(int i=0;i<=a;i++)
    {
        c+=i;
        if(c == b)
        {
            c=c-1;
        }
    }
    printf("%ld", c % 1000000007);
    return 0;
}
```

	Input	Expected	Got	
✓	2 2	3	3	✓
✓	2 1	2	2	✓
✓	3 3	5	5	✓

Passed all tests! ✓

Falling back to raw text area.

```
#include<stdio.h>
int main()
{
    int a,b;
    scanf("%d",&a);
    scanf("%d", &b);

    int c[a],d=0;
    for(int i=1;i<=a;i++)
    {
        if((a%i)==0)
        {
            c[d]=i;
            d++;
            if(d>=b)
            {
                break;
            }
```

	Input	Expected	Got	
✓	10 3	5	5	✓
✓	10 5	0	0	✓
✓	1 1	1	1	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
int t,k;
scanf("%d",&t);
while(t--)
{
int n;
scanf("%d",&n);
int a[n];
for(int i=0;i<n;i++)
{
scanf("%d",&a[i]);
}
scanf("%d",&k);
int flag=0;
for(int i=0;i<n;i++)
{
for(int j=i+1;j<n;j++)
{
if(a[i]-a[j]==k||a[j]-a[i]==k)
{
flag=1;
break;
}
}
if(flag)
break;
}
printf("%d\n",flag);
}
}
```

	Input	Expected	Got	
✓	1 3 1 3 5 4	1	1	✓
✓	1 3 1 3 5 99	0	0	✓

Passed all tests! ✓

ion **2**

ct

ed out of

Sam loves chocolates and starts buying them on the 1st day of the year. I
odd, Sam will buy x chocolates; on days when x is even, Sam will not purc

Complete the code in the editor so that for each day Ni (where 1 ≤ x ≤ N

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        int n,c=0;
        scanf("%d",&n);
        for(int i=0;i<=n;i++)
        {
            if(i%2!=0)
            c=c+i;
        }
        printf("%d\n",c);
    }
}
```

	Input	Expected	Got	
✓	3 1 2 3	1 1 4	1 1 4	✓
✓	10 71 100 86 54 40 9 77 9 13 98	1296 2500 1849 729 400 25 1521 25 49 2401	1296 2500 1849 729 400 25 1521 25 49 2401	✓

Thus, the function returns the array [1, 0, 3, 4] as the answer.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d",&a);
    int ta[a];
    for(int i=0;i<a;i++)
    scanf("%d",&ta[i]);
    scanf("%d",&b);
    int tb[b];
    for(int i=0;i<b;i++)
    scanf("%d",&tb[i]);
    for(int j=0;j<b;j++)
    {
        c=0;
        for(int i=0;i<a;i++)
        {
            if(tb[j]>=ta[i])
            c++;

        }
        printf("%d\n",c);
    }
}
```

	Input	Expected	Got	
✓	4 1 4 2 4 2 3 5	2 4	2 4	✓
✓	5 2 10 5 4 8 4 3 1 7 8	1 0 3 4	1 0 3 4	✓

Passed all tests! ✓

3	

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<limits.h>
int main()
{
    int a;
    scanf("%d",&a);
    int b[a];
    for(int i=0;i<a;i++)
    {
        scanf("%d",&b[i]);
    }
    int c;
    scanf("%d",&c);
    for(int i=0;i<a-c+1;i++)
    {
        int m=INT_MIN;
        for(int j=0;j<c;j++)
        {
            if(b[i+j]>m)
            {
                m=b[i+j];
            }
        }
        printf("%d ",m);
    }
}
```

```
            {
                m=b[i+j];
            }
        }
        printf("%d ",m);
    }
}
```

	Input	Expected	Got	
✓	8 1 3 5 2 1 8 6 9 3	5 5 5 8 8 9	5 5 5 8 8 9	✓
✓	10 3 7 5 1 2 9 8 5 3 2 3	7 7 5 9 9 9 8 5	7 7 5 9 9 9 8 5	✓

Passed all tests! ✓

estion **2**

rrect

rked out of

Given an array and a threshold value find the output.

Input: {5,8,10,13,6,2}

Threshold = 3

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{
    int a;
    scanf("%d",&a);
    int b[a];
    for(int i=0;i<a;i++)
    scanf("%d",&b[i]);
    int c;
    scanf("%d",&c);
    int d=0;
    for(int i=0;i<a;i++)
    {
        int e=b[i];
        for(int j=c;j>0;j--)
        {
            while(e>=j)
            {
                e=e-j;
                d++;
            }
        }
    }
    printf("%d ",d);
}
```

```
    }
    printf("%d ",d);
}
```

	Input	Expected	Got	
✓	6 5 8 10 13 6 2 3	17	17	✓
✓	7 20 35 57 30 56 87 30 10	33	33	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 1.00

Flag question

Output is a merged array without duplicates.

Input Format

N1 - no of elements in array 1

Array elements for array 1

```c
#include<stdio.h>
#include<stdlib.h>
int cmp(const void*a,const void*b)
{
    return(*((int*)a)- *((int*)b));
}
int main()
{
    int a,b;
    scanf("%d",&a);
    int c[a];
    for(int i=0;i<a;i++)
    {
        scanf("%d",&c[i]);
    }
    scanf("%d",&b);
    int d[b];
    for(int i=0;i<b;i++)
    {
        scanf("%d",&d[i]);
    }
    int f=a+b;
    int e[f];
    int k=0;
    for (int i=0;i<a;i++)
    {
        e[k++]=c[i];
    }
    for(int i=0;i<b;i++)
    {
        int s=0;
        for(int j=0;j<a;j++)
        {
            if(d[i]==c[j])
            {
                s=1;
                break;
```

```
            {
                s=1;
                break;
            }
        }
        if(!s)
    {
        e[k]=d[i];
        k++;
    }
}
qsort(e,k,sizeof(int),cmp);
for(int i=0;i<k;i++)
{
    printf("%d ",e[i]);
}
}
```

	Input	Expected	Got	
✓	5 1 2 3 6 9 4 2 4 5 10	1 2 3 4 5 6 9 10	1 2 3 4 5 6 9 10	✓

Passed all tests! ✓

```c
#include<stdio.h>
int main()
{
    int a;
    scanf("%d",&a);
    while(a--)
    {
        int b,c;
        scanf("%d %d",&b,&c);
        int d[c];
        for(int i=0;i<c;i++)
        {
            scanf("%d",&d[i]);
        }
        int s=0;
        for(int i=0;i<c;i++)
        {
            for(int j=0;j<c;j++)
            {
                if(i==j)
                {
                    continue;
                }
                if((d[i]+d[j])==b)
                {
                    printf("%d %d\n",(i+1),(j+1));
                    s=1;
                    break;
                }
            }
            if(s==1)
            {
                break;
            }
        }
    }
}
```

```
37 }
```

	Input	Expected	Got	
✓	2 4 5 1 4 5 3 2 4 4 2 2 4 3	1 4 1 2	1 4 1 2	✓

Passed all tests! ✓

Question **2**

Correct

Marked out of 5.00

Numeros the Artist had two lists that were permutations of one another. He was ve
exhibition to another, some numbers were lost out of the first list. Can you find the

```c
#include<stdio.h>
#include<stdlib.h>
int cmp(const void*a,const void*b)
{
    return (*(int*)a-*(int*)b);
}
int main()
{
    int x[10000]={0};
    int y[10000]={0};
    int z[10000]={0};
    int a;
    scanf("%d",&a);
    int b[a];
    for(int i=0;i<a;i++)
    {
        scanf("%d",&b[i]);
        x[b[i]]++;
    }
    int c;
    scanf("%d",&c);
    int d[c];
    for(int j=0;j<c;j++)
    {
        scanf("%d",&d[j]);
        y[d[j]]++;
    }
    qsort(d,c,sizeof(int),cmp);
    for(int i=0;i<c;i++)
    {
        if(z[d[i]]==0)
        {
            if(y[d[i]]!=x[d[i]])
            {
                int q=y[d[i]]-x[d[i]];
                for(int j=0;j<q;j++)
                {
                    printf("%d ",d[i]);
```

```
    int c;
    scanf("%d",&c);
    int d[c];
    for(int j=0;j<c;j++)
    {
        scanf("%d",&d[j]);
        y[d[j]]++;
    }
    qsort(d,c,sizeof(int),cmp);
    for(int i=0;i<c;i++)
    {
        if(z[d[i]]==0)
        {
            if(y[d[i]]!=x[d[i]])
            {
                int q=y[d[i]]-x[d[i]];
                for(int j=0;j<q;j++)
                {
                    printf("%d ",d[i]);
                }
            }
            z[d[i]]=1;
        }
    }
}
```

```
                printf("%d ",d[i]);
            }
        }
        z[d[i]]=1;
    }
}
}
```

	Input	Expected	Got	
✓	10 203 204 205 206 207 208 203 204 205 206 13 203 204 204 205 206 207 205 208 203 206 205 206 204	204 205 206	204 205 206	✓

Passed all tests! ✓

Watson gives Sherlock an array of integers. His challenge is to find an element of the array such that the sum o
equal to the sum of all elements to the right. For instance, given the array ***arr = [5, 6, 8, 11]***, ***8*** is between two
your starting array is ***[1]***, that element satisfies the rule as left and right sum to ***0***.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a;
    scanf("%d",&a);
    while(a--)
    {
        int b;
        scanf("%d",&b);
        int c[b];
        for(int i=0;i<b;i++)
        {
            scanf("%d",&c[i]);
        }
        int x=0;
        for(int i=0;i<b;i++)
        {
            int s1=0,s2=0;
            for(int j=0;j<i;j++)
            {
                s1+=c[j];
            }
            for(int j=i+1;j<b;j++)
            {
                s2+=c[j];
            }
            if(s1==s2)
            {
                x=1;
                break;
            }
        }
        (x==1)?printf("YES\n"):printf("NO\n");
    }
```

Search

```
            {
                x=1;
                break;
            }
        }
        (x==1)?printf("YES\n"):printf("NO\n");
    }
}
```

	Input	Expected	Got	
✓	3 5 1 1 4 1 1 4 2 0 0 0 4 0 0 2 0	YES YES YES	YES YES YES	✓
✓	2 3 1 2 3 4 1 2 3 3	NO YES	NO YES	✓

Passed all tests! ✓

```c
#include<stdio.h>
#include<stdlib.h>
#include<limits.h>
int cmp(const void*a,const void*b)
{
    return (*(int*)a-*(int*)b);
}
int main()
{
    int a,b,c;
    scanf("%d",&a);
    while(a--)
    {
        scanf("%d%d",&b,&c);
        int d[b];
        for(int i=0;i<b;i++)
        {
            scanf("%d",&d[i]);
        }
        qsort(d,b,sizeof(int),cmp);
        int mas=0;
        for(int i=c;i<b;i++)
        {
            mas+=d[i];
        }

        int mis=0;
        for(int i=0;i<b-c;i++)
        {
         mis+=d[i];
        }
        printf("%d\n",(mas-mis));
    }
}
```

```
            mis+=d[i];
        }
        printf("%d\n",(mas-mis));
    }
}
```

	Input	Expected	Got	
✓	1 5 1 1 2 3 4 5	4	4	✓

Passed all tests! ✓

Question **2**

Correct

Marked out of 1.00

Flag question

A new deadly virus has infected large population of a planet. A brilliant
disease. Vaccine produced from this virus has various strength dependi
count in vaccine batch is more than midichlorians count of person. A d
of each infected patient, Practo stores all vaccine doctor has and their
patients with the vaccines he has. The number of vaccines and patients

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a;
    scanf("%d",&a);
    int b[a],c[a];
    for(int i=0;i<a;i++)
    {
        scanf("%d",&b[i]);
    }
    for(int i=0;i<a;i++)
    {
        scanf("%d",&c[i]);
    }
    int s=1;
    for(int i=0;i<a;i++)
    {
        if(c[i]<b[i])
        {
            s=0;
        }
    }
    (s==1)?printf("Yes"):printf("No");
}
```

	Input	Expected	Got	
✓	5 123 146 454 542 456 100 328 248 689 200	No	No	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a;
    scanf("%d",&a);
    int b[a];
    for(int i=0;i<a;i++)
    {
        scanf("%d",&b[i]);
    }
    int c=0;
    for(int i=0;i<a;i++)
    {
        for(int j=i+1;j<a;j++)
        {
            if(b[i]==b[j])
            {
                c++;
            }
        }
    }
    printf("%d",c);
}
```

	Input	Expected	Got	
✓	5 1 3 1 4 3	2	2	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<stdlib.h>
int cmp(const void*a,const void*b)
{
    return (*(int*)a-*(int*)b);
}
int main()
{
    int a;
    scanf("%d",&a);
    int b[a],c[a];
    for(int i=0;i<a;i++)
    {
        scanf("%d",&b[i]);
        c[i]=b[i];
    }
    qsort(c,a,sizeof(int),cmp);
    for(int i=0;i<a;i++)
    {
        for(int j=0;j<a;j++)
        {
            if(c[i]==b[j])
            {
                printf("%d ",j);
            }
        }
    }
}
```

```c
    qsort(c,a,sizeof(int),cmp);
    for(int i=0;i<a;i++)
    {
        for(int j=0;j<a;j++)
        {
            if(c[i]==b[j])
            {
                printf("%d ",j);
            }
        }
    }
}
```

	Input	Expected	Got	
✓	5 4 5 3 7 1	4 2 0 1 3	4 2 0 1 3	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a[3][3];
    int s1 =0,s2 = 0,s = 0;
    for(int i=0;i<3;i++)
    {
        for(int j=0;j<3;j++)
        {
            scanf("%d",&a[i][j]);
            if(s == 0)
            {
                s1 += a[i][j];
                s =1;
            }
            else
            {
                s2 +=a[i][j];
                s=0;
            }
        }
    }
    printf("%d\n%d",s1,s2);


}
```

```
            }
            else
            {
                s2 +=a[i][j];
                s=0;
            }
        }
    }
    printf("%d\n%d",s1,s2);


}
```

	Input	Expected	Got	
✓	1 2 3 4 5 6 7 8 9	25 20	25 20	✓
✓	21 422 423 443 586 645 657 846 904	2591 2356	2591 2356	✓

Passed all tests! ✓

Question **2**

Microsoft has come to hire interns from your college. N students got shortlisted out of which few
levels. Smaller the talent level, lesser is your chance to be selec

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<stdlib.h>

int com(const void *a, const void *b)
{
    return (*(int*)b - *(int*)a);
}
int main()
{
    int a;
    scanf("%d",&a);

    int b[a][2], c[a], d[a],x=0,y=0;
    for (int i =0;i < a;i ++)
    {
        scanf("%d", &b[i][0]);
        scanf("%d", &b[i][1]);

        if(b[i][0] == 0)
        {
            c[x]=b[i][1];
            x++;
        }
        else
        {
            d[y] = b[i][1];
            y++;
        }
    }
    qsort(c, x, sizeof(int), com);
    qsort(d, y, sizeof(int), com);

    for(int i = 0; i < x; i ++)
    {
        printf("%d ",c[i]);
    }
```

```
            d[y] = b[i][1];
            y++;
        }
    }
    qsort(c, x, sizeof(int), com);
    qsort(d, y, sizeof(int), com);

    for(int i = 0; i < x; i ++)
    {
        printf("%d ",c[i]);
    }
    for(int i = 0;i < y; i++)
    {
        printf("%d ", d[i]);

    }
    return 0;

}
```

	Input	Expected	Got	
✓	5 0 3 1 6 0 2 0 7 1 15	7 3 2 15 6	7 3 2 15 6	✓
✓	6 0 1 0 26 0 39 0 37 0 7 0 13	39 37 26 13 7 1	39 37 26 13 7 1	✓
✓	12 1 12 1 14 1 18 1 1 1 2 1 3 1 5 1 8 1 9 1 10 0 29 0 31	31 29 18 14 12 10 9 8 5 3 2 1	31 29 18 14 12 10 9 8 5 3 2 1	✓
✓	12 0 12 1 12	12 12 12 12 12 12 12 12 12 12 12 12	12 12 12 12 12 12 12 12 12 12 12 12	✓

	1 10 0 29 0 31			
✓	12 0 12 1 12 0 12 1 12 0 12 0 12 1 12 0 12 1 12 1 12 0 12 1 12	12 12 12 12 12 12 12 12 12 12 12 12	12 12 12 12 12 12 12 12 12 12 12 12	✓

Passed all tests! ✓

on **3**

t

d out of

question

Shyam Lal, a wealthy landlord from the state of Rajasthan, being an old fellow and tired of doing hard work, decided to se and to live rest of his life with that money. No other farmer is rich enough to buy all his land so he decided to partition the rectangular plots of different sizes with different cost per unit area. So, he sold these plots to the farmers but made a mist he made partitions that could be overlapping. When the farmers came to know about it, they ran to him for compensatior they paid to him. So, he decided to return all the money to the farmers of that land which was overlapping with other farn down the conflict. All the portion of conflicted land will be taken back by the landlord.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int i,j,n,x1,x2,y1,y2,t=0;
    long long total = 0;
    int arr[1001][1001] = {0};

    scanf("%d",&n);
    while(n--)
    {
        scanf("%d %d %d %d %d",&x1, &y1, &x2, &y2, &t);

        for(i = x1;i<= x2; i++)
        {
            for(j = y1; j<=y2; j++)
            {
                if(arr[i][j] == 0)
                    arr[i][j] +=t;
                else if(arr[i][j] > 0)
                    arr[i][j] = (-1) * (arr[i][j] + t);
                else if(arr[i][j] < 0)
                    arr[i][j] -= t;
            }
        }
    }
    for(i=1;i<1001;i++)
    {
        for(j = 1; j< 1001; j ++)
        {
            if(arr[i][j] < 0)
                total += arr[i][j];
        }
    }
    printf("%lld", (-1) * total);
    return 0;
}
```

```
    }
    printf("%lld", (-1) * total);
    return 0;
}
```

	Input	Expected	Got	
✓	3 1 4 4 6 1 4 3 6 6 2 2 2 5 4 3	35	35	✓
✓	1 48 12 49 27 8	0	0	✓
✓	3 88 34 99 76 44 82 65 94 100 81 58 16 65 66 7	10500	10500	✓

Passed all tests! ✓

- ***1*** occurs two times.
- ***2, 4, 5, 6*** and ***7*** occur one time each.

The remaining digits ***0, 3, 8*** and ***9*** don't occur at all.

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int main()
{   int s[10] = {0};
    char a[10000];
    scanf("%s", a);

    for(int i = 0; i < 10; i++)
    {
        char b = '0' + i;
        int c = 0;
        for(int j = 0; a[j]; j++)
        {
            if(a[j] == b)
            {
                c++;
            }
        }
        s[i] = c;
    }

    for(int i = 0; i < 10; i++)
    {
        printf( "%d ",s[i]);
    }
}
```

```
        {
            printf( "%d ",s[i]);
        }
}
```

	Input	Expected	Got	
✓	a11472o5t6	0 2 1 0 1 1 1 1 0 0	0 2 1 0 1 1 1 1 0 0	✓
✓	lw4n88j12n1	0 2 1 0 1 0 0 0 2 0	0 2 1 0 1 0 0 0 2 0	✓
✓	1v888861256338ar0ekk	1 1 1 2 0 1 2 0 5 0	1 1 1 2 0 1 2 0 5 0	✓

Passed all tests! ✓

estion **2**

rrect

arked out of
0

Flag question

Today, Monk went for a walk in a garden. There are many trees in the garden and each
walking, he noticed that all trees with vowels on it are not in good state. He decided to
count of such trees in the garden.

In test case 1, a and o are the only vowels. So, count=2

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{
    int d;
    scanf("%d",&d);
    while(d--)
    {
        char a[100001],c=0;
        scanf("%s",a);
        char b[10] = {'A', 'E', 'I', 'O', 'U', 'a', 'e', 'i', 'o', 'u'};
        for(int i=0;a[i] !='\0';i++)
        {
            for(int j=0;j<10;j++)
            {
                if(a[i] == b[j])
                {
                    c++;
                }
            }
        }
        printf("%d\n",c);
    }
    return 0;
}
```

```
    }
    return 0;
}
```

	Input	Expected	Got	
✓	2 nBBZLaosnm JHkIsnZtTL	2 1	2 1	✓
✓	2 nBBZLaosnm JHkIsnZtTL	2 1	2 1	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of

Given a sentence, ***s***, print each word of the sentence in a new line.

In the given string, there are three words ["This", "is", "C"]. We have to print each of thes

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<string.h>

int main()
{
    char a[100];
    int i = 0;

    fgets(a, sizeof(a),stdin);
    while(a[i] !='\0')
    {
        char b[100];
        int j = 0;

        while (a[i] !=' ' && a[i] != '\0')
        {
            b[j] = a[i];
            i++;
            j++;

        }
        b[j] = '\0';
        printf("%s\n",b);
        if(a[i] == ' ')
        {
            i++;
        }
    }
    return 0;

}
```

```
        }
    }
    return 0;

}
```

	Input	Expected	Got	
✓	This is C	This is C	This is C	✓
✓	Learning C is fun	Learning C is fun	Learning C is fun	✓

Passed all tests! ✓

Question **4**

Correct

Input Format

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    char a[100],b[100];
    scanf("%s", a);
    scanf("%s", b);

    int i=0,j=0;

    while(a[i] !='\0')
    {
        i++;
    }
    while(b[j] != '\0')
    {
        j++;
    }
    printf("%d %d\n%s%s\n",i,j,a,b);
    char c=b[0];
    b[0] = a[0];
    a[0] = c;

    printf("%s %s",a,b);
}
```

```
    }
    while(b[j] != '\0')
    {
        j++;
    }
    printf("%d %d\n%s%s\n",i,j,a,b);
    char c=b[0];
    b[0] = a[0];
    a[0] = c;

    printf("%s %s",a,b);
}
```

	Input	Expected	Got	
✓	abcd ef	4 2 abcdef ebcd af	4 2 abcdef ebcd af	✓

Passed all tests! ✓

```c
#include<stdio.h>
#include<string.h>
#include<stdbool.h>

bool areCompatible(char a[], char b[])
{
    int lenA = strlen(a);
    int lenB = strlen(b);

    if ( lenA != lenB)
    {
        return false;
    }
    for (int i = 0; i < lenA; i++)
    {
        if(a[i] > b[i])
        {
            return false;
        }
    }
    return true;
}
int main()
{
    char a[100001];
    char b[100001];

    scanf("%s", a);
    scanf("%s", b);

    if(areCompatible(a,b))
    {
        printf("YES\n");
    }
    else
    {
        printf("NO\n");
    }
```

```c
    char a[100001];
    char b[100001];

    scanf("%s", a);
    scanf("%s", b);

    if(areCompatible(a,b))
    {
        printf("YES\n");
    }
    else
    {
        printf("NO\n");
    }
    return 0;
}
```

	Input	Expected	Got	
✓	abaca cdbda	YES	YES	✓

Passed all tests! ✓

```c
#include<stdio.h>
#include<string.h>
#include<stdlib.h>

void rev(char c[], const char a[])
{
    int len = strlen(a);
    for(int i =0 ; i < len; i++)
    {
        c[i] = a[len - i - 1];

    }
    c[len] = '\0';
}
int che(const char a[], const char b[])
{
    return strcmp(a,b);
}
int main()
{
    int a;
    scanf("%d", &a);
    char **b = (char **)malloc(a * sizeof(char *));
    for(int i=0; i < a; i++)
    {
        b[i] = (char *)malloc(100 * sizeof(char));
        scanf("%s",b[i]);

    }
    for (int i=0; i<a; i++)
    {
        char c[100];
        rev(c,b[i]);
        for(int j=0;j<a;j++)
        {
            if(che(c, b[j]) == 0)
            {
                printf("%ld %c\n", strlen(b[j]), b[i][strlen(b[j]) / 2]);
                return 0;
```

```
    {
        char c[100];
        rev(c,b[i]);
        for(int j=0;j<a;j++)
        {
            if(che(c, b[j]) == 0)
            {
                printf("%ld %c\n", strlen(b[j]), b[i][strlen(b[j]) / 2]);
                return 0;
            }
        }
    }
    return 0;
}
```

	Input	Expected	Got	
✓	4 abc def feg cba	3 b	3 b	✓

Passed all tests! ✓

```c
#include<stdio.h>
#include<string.h>
#include<stdlib.h>

typedef struct
{
    char name[21];
    int points;
} Restaurant;

int compare(const void *a, const void *b)
{
    Restaurant *restA = (Restaurant*)a;
    Restaurant *restB = (Restaurant*)b;

    if (restA->points != restB->points)
    {
        return restB->points - restA->points;

    }
    else
    {
        return strcmp( restA->name, restB->name);
    }
}
int main()
{
    int n;
    scanf("%d", &n);

    Restaurant *restaurants = (Restaurant *)malloc(n * sizeof(Restaurant));
    if (!restaurants)
    {
        printf("Memory allocation failed.\n");
        return 1;
    }
    for(int i=0;i<n;i++)
    {
        scanf("%s %d", restaurants[i].name, & restaurants[i].points);
```

```c
        printf("Memory allocation failed.\n");
        return 1;
    }
    for(int i=0;i<n;i++)
    {
        scanf("%s %d", restaurants[i].name, & restaurants[i].points);

    }
    qsort(restaurants, n, sizeof(Restaurant),compare);

    printf("%s\n", restaurants[0].name);
    free(restaurants);

    return 0;
}
```

	Input	Expected	Got	
✓	3 Pizzeria 108 Dominos 145 Pizzapizza 49	Dominos	Dominos	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<string.h>
#include<ctype.h>

void check_mobile_number(char s[])
{
    if (strlen(s) !=10 || !isdigit(s[0]) || s[0] == '0')
    {
        printf("NO\n");
        return;
    }
    for (int i=0; i<10;i++)
    {
        if(!isdigit(s[i]))
        {
            printf("NO\n");
            return;
        }
    }
    printf("YES\n");

}
int main()
{
    int t;
    scanf("%d", &t);
    char s[20];
    for(int i=0;i < t;i++)
    {
        scanf("%s", s);
        check_mobile_number(s);

    }
    return 0;
}
```

```
    {
        scanf("%s", s);
        check_mobile_number(s);

    }
    return 0;
}
```

	Input	Expected	Got	
✓	3 1234567890 0123456789 0123456.87	YES NO NO	YES NO NO	✓

Passed all tests! ✓

Search

```c
/*
 * Complete the 'fourthBit' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER number as parameter.
 */

int fourthBit(int number)

{
    return (number >> 3) & 1;
}
```

	Test	Expected	Got	
✓	printf("%d", fourthBit(32))	0	0	✓
✓	printf("%d", fourthBit(77))	1	1	✓

Passed all tests! ✓

```c
/*
 * Complete the 'pthFactor' function below.
 *
 * The function is expected to return a LONG_INTEGER.
 * The function accepts following parameters:
 *  1. LONG_INTEGER n
 *  2. LONG_INTEGER p
 */

long pthFactor(long n, long p)
{
    long j=0;
    for(long i=1;i<=n;i++)
    {
        if(n%i == 0)
        {
            j++;
            if(j==p)
            {
                return i;
            }
        }
    }
    return 0;
}
```

	Test	Expected	Got	
✓	printf("%ld", pthFactor(10, 3))	5	5	✓
✓	printf("%ld", pthFactor(10, 5))	0	0	✓
✓	printf("%ld", pthFactor(1, 1))	1	1	✓

Answer: (penalty regime: 0 %)

Reset answer

```
/*
 * Complete the 'myFunc' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER n as parameter.
 */

int myFunc(int n)
{
    while(n>1)
    {
        if(n%20==0)
        {
            n/=20;
        }
        else if(n%10==0)
        {
            n/=10;
        }
        else
        {
            return 0;
        }
    }
    return 1;
}
```

	Test	Expected	Got

```
            n/=10;
        }
        else
        {
            return 0;
        }
    }
    return 1;
}
```

	Test	Expected	Got	
✓	printf("%d", myFunc(1))	1	1	✓
✓	printf("%d", myFunc(2))	0	0	✓
✓	printf("%d", myFunc(10))	1	1	✓
✓	printf("%d", myFunc(25))	0	0	✓
✓	printf("%d", myFunc(200))	1	1	✓

Passed all tests! ✓

Question **2**

Find the number of ways that a given integer, ***X***, can be expressed as the sum of the ***N***th powers

Reset answer

```c
/*
 * Complete the 'powerSum' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts following parameters:
 *  1. INTEGER x
 *  2. INTEGER n
 */

int powerSum(int x, int m, int n)
{
    if(x==0)
    {
        return 1;
    }
    if(x<0)
    {
        return 0;
    }
    int count=0;
    for(int i=m;;i++)
    {
        int power=1;
        for(int j=0;j<n;j++)
        {
            power*=i;
        }
        if(power>x)
        {
            break;
        }
        count+=powerSum(x-power,i+1,n);

    }
    return count;
}
```

```
            break;
        }
        count+=powerSum(x-power,i+1,n);
    }
    return count;
}
```

	Test	Expected	Got	
✓	`printf("%d", powerSum(10, 1, 2))`	1	1	✓

Passed all tests! ✓

Search

Reset answer

```c
/*
 * Complete the 'balancedSum' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER_ARRAY arr as parameter.
 */

int balancedSum(int arr_count, int* arr)
{
    int l=0,r=0;
    for(int i=0;i<arr_count;i++)
    {
        r+=arr[i];
    }
    for(int i=0;i<arr_count;i++)
    {
        if(l==r - arr[i])
        {
            return i;
        }
        l+= arr[i];
        r-= arr[i];
    }
    return 1;
}

```

	Test	Expected	Got	
✓	int arr[] = {1,2,3,3}; printf("%d", balancedSum(4, arr))	2	2	✓

Answer: (penalty regime: 0 %)

Reset answer

```c
/*
 * Complete the 'arraySum' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER_ARRAY numbers as parameter.
 */

int arraySum(int numbers_count, int *numbers)
{
    int s =0;
    for(int i=0;i<numbers_count;i++)
    {
        s+=numbers[i];
    }
    return s;
}

```

	Test	Expected	Got	
✓	`int arr[] = {1,2,3,4,5};` `printf("%d", arraySum(5, arr))`	15	15	✓

Passed all tests! ✓

Search

Reset answer

```c
/*
 * Complete the 'minDiff' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER_ARRAY arr as parameter.
 */

int minDiff(int arr_count, int* arr)
{
    for(int i=0;i<arr_count;i++)
    {
        for(int j=i;j<arr_count;j++)
        {
            if(i!=j)
            {
                if(arr[i] > arr[j])
                {
                    int temp = arr[j];
                    arr[j] = arr[i];
                    arr[i] = temp;
                }
            }
        }
    }
    int m=0;
    for(int i=0;i<arr_count-1;i++)
    {
        m+=arr[i+1] - arr[i];
    }
    return m;
}
```

```
    int m=0;
    for(int i=0;i<arr_count-1;i++)
    {
        m+=arr[i+1] - arr[i];
    }
    return m;
}
```

	Test	Expected	Got	
✓	`int arr[] = {5, 1, 3, 7, 3};` `printf("%d", minDiff(5, arr))`	6	6	✓

Passed all tests! ✓

Search

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
struct Box
{
    int length;
    int width;
    int height;
};
int main()
{
    int n;
    scanf("%d",&n);
    struct Box boxes[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d %d %d",&boxes[i].length, &boxes[i].width,&boxes[i].height);
        if(boxes[i].height < 41)
        {
            printf("%d\n",boxes[i].length * boxes[i].width * boxes[i].height);
        }
    }
    return 0;
}
```

	Input	Expected	Got	
✓	4 5 5 5 1 2 40 10 5 41 7 2 42	125 80	125 80	✓

```c
#include<stdio.h>
#include<math.h>
struct Triangle
{
    int a,b,c;
    double area;
};
int main()
{
    int n;
    scanf("%d",&n);
    struct Triangle triangles[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d %d %d",&triangles[i].a,&triangles[i].b,&triangles[i].c);
        double p=(triangles[i].a + triangles[i].b+triangles[i].c)/2.0;
        triangles[i].area = sqrt(p*(p-triangles[i].a) *(p-triangles[i].b)*(p-triangles[i].c));
    }
    for(int i=0;i<n-1;i++)
    {
        for(int j=i+1;j<n;j++)
        {
            if(triangles[i].area > triangles[j].area)
            {
                struct Triangle temp =triangles[i];
                triangles[i]=triangles[j];
                triangles[j]=temp;

            }
        }
    }
    for(int i=0;i<n;i++)
    {
        printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
    }
}
```

```
                triangles[j]=temp;

            }
        }
    }
    for(int i=0;i<n;i++)
    {
        printf("%d %d %d\n",triangles[i].a,triangles[i].b,triang
    }
}
```

	Input	Expected	Got	
✓	3 7 24 25 5 12 13 3 4 5	3 4 5 5 12 13 7 24 25	3 4 5 5 12 13 7 24 25	✓

Passed all tests! ✓

```c
/*
 * Complete the 'reverseArray' function below.
 *
 * The function is expected to return an INTEGER_ARRAY.
 * The function accepts INTEGER_ARRAY arr as parameter.
 */

/*
 * To return the integer array from the function, you should:
 *     - Store the size of the array to be returned in the result_count variable
 *     - Allocate the array statically or dynamically
 *
 * For example,
 * int* return_integer_array_using_static_allocation(int* result_count) {
 *     *result_count = 5;
 *
 *     static int a[5] = {1, 2, 3, 4, 5};
 *
 *     return a;
 * }
 *
 * int* return_integer_array_using_dynamic_allocation(int* result_count) {
 *     *result_count = 5;
 *
 *     int *a = malloc(5 * sizeof(int));
 *
 *     for (int i = 0; i < 5; i++) {
 *         *(a + i) = i + 1;
 *     }
 *
 *     return a;
 * }
 *
 */
int* reverseArray(int arr_count, int *arr, int *result_count)
{
      *result_count = arr_count;
      static int rev[100];
```

```
 *
 *     return a;
 * }
 *
 * int* return_integer_array_using_dynamic_allocation(int* result_count) {
 *     *result_count = 5;
 *
 *     int *a = malloc(5 * sizeof(int));
 *
 *     for (int i = 0; i < 5; i++) {
 *         *(a + i) = i + 1;
 *     }
 *
 *     return a;
 * }
 *
 */
int* reverseArray(int arr_count, int *arr, int *result_count)
{
    *result_count = arr_count;
    static int rev[100];
    int i,j=0;
    for(i=arr_count-1;i>=0;i--)
    rev[j++] = arr[i];
    return rev;
}
```

```
41        for (i = arr_count-1;i>=0;i--)
42            rev[j++] = arr[i];
43        return rev;
44    }
```

	Test	Expected	Got	

✓	`int arr[] = {1, 3, 2, 4, 5};`
	`int result_count;`
	`int* result = reverseArray(5, arr, &result_count);`
	`for (int i = 0; i < result_count; i++)`
	`printf("%d\n", *(result + i));`
		5
		4
		2
		3
		1
			5
			4
			2
			3
			1
				✓

Passed all tests! ✓

An automated cutting machine is used to cut rods into segments. The cutting machine can only hold
only make one cut at a time. Given the array *lengths[]* representing the desired lengths of each segme
the necessary cuts using this machine. The rod is marked into lengths already, in the order given.

Reset answer

```c
/*
 * Complete the 'cutThemAll' function below.
 *
 * The function is expected to return a STRING.
 * The function accepts following parameters:
 *  1. LONG_INTEGER_ARRAY lengths
 *  2. LONG_INTEGER minLength
 */

/*
 * To return the string from the function, you should either do static allocation or dynamic allocation
 *
 * For example,
 * char* return_string_using_static_allocation() {
 *     static char s[] = "static allocation of string";
 *
 *     return s;
 * }
 *
 * char* return_string_using_dynamic_allocation() {
 *     char* s = malloc(100 * sizeof(char));
 *
 *     s = "dynamic allocation of string";
 *
 *     return s;
 * }
 *
 */
char* cutThemAll(int lengths_count, long *lengths, long minLength)
{
    int s=0;
    for(int i=0;i<lengths_count-1;i++)
    {
        s+= *(lengths + i);
    }
```

```c
{
    int s=0;
    for(int i=0;i<lengths_count-1;i++)
    {
        s+= *(lengths + i);
    }
    if(s >= minLength)
    {
        return "Possible";
    }
    else
    {
        return "Impossible";
    }
}
```

	Test	Expected	Got	
✓	`long lengths[] = {3, 5, 4, 3};` `printf("%s", cutThemAll(4, lengths, 9))`	Possible	Possible	✓
✓	`long lengths[] = {5, 6, 2};` `printf("%s", cutThemAll(3, lengths, 12))`	Impossible	Impossible	✓

Passed all tests! ✓