

Problem No: 2302056_101

Problem Name: There are three given ranges. Write a C program that reads a floating-point number and finds the range where it belongs from four given ranges.

Input:

```
1  #include<stdio.h>
2  int main()
3
4  ... int x;
5  ... printf("Enter the number: ");
6  ... scanf("%d",&x);
7  ... if(x>=0 && x<=30) printf("0-30");
8  ... if(x>=31 && x<=50) printf("0-30");
9  ... if(x>=51 && x<=80) printf("0-30");
10 ... if(x>=81 && x<=100) printf("0-30");
11 ... return 0;
12 }
```

Output:

```
Enter the number: 25
0-30
```

Problem No: 2302056_103

Problem Name: In science, a multiple is the product of any quantity and an integer. In other words, for the quantities a and b, we say that b is a multiple of a if $b = na$ for some integer n, which is called the multiplier. If a is not zero, this is equivalent to saying that b/a is an integer.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int x,y;
6      printf("Enter x and y: \n");
7      scanf("%d",&x);
8      scanf("%d",&y);
9      if(x%y==0 || y%x==0){
10         printf("Multiplies");
11     } else{
12         printf("No");
13     }
14     return 0;
15 }
```

Output:

```
Enter x and y:
```

```
10
```

```
5
```

```
Multiplies
```

Problem No: 2302056_105

Problem Name: Write a C program that accepts seven floating point numbers and counts the number of positive and negative numbers. Print the average of all positive and negative values with two digits after the decimal number.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      printf("Enter the number(under 7): \n");
6      float x;
7      int count_minus = 0;
8      float sum_minus = 0;
9      int count_plus = 0;
10     float sum_plus = 0;
11     for(int i=1; i<=7; i++){
12         scanf("%f",&x);
13         if(x>0){
14             sum_plus = sum_plus + x;
15             count_plus++;
16         }
17         if(x<0){
18             sum_minus = sum_minus + x;
19             count_minus++;
```

```

20     ....}
21     ....}
22     ....float avrg_plus = sum_plus/count_plus;
23     ....float avrg_minus = sum_minus/count_minus;
24
25     ....printf("\n%d Number of positive numbers:avarage = %.2f\n ",count_plus,avrg_plus);
26     ....printf("\n%d Number of negative numbers:avarage = %.2f\n ",count_minus,avrg_minus);
27     ....return 0;
28 }

```

Output:

```

Enter the number(under 7):
5
1
2
3
5
4
8

7 Number of positive numbers:avarage = 4.00
0 Number of negative numbers:avarage = -1.#J

```

Problem No: 2302056_06

Problem Name: Write a C program that accepts 7 integer values and counts the even, odd, positive and negative values.

Input:

```
1  #include<stdio.h>
2  int main()
3  {
4      ...int arr[7];
5      ...int positive=0;
6      ...int negative=0;
7      ...int even=0;
8      ...int odd=0;
9      ...for(int i=0; i<7; i++){
10         ...scanf("%d",&arr[i]);
11         ...if(arr[i]>0) positive++;
12         ...if(arr[i]<0) negative++;
13         ...if(arr[i]%2==0) even++;
14         ...if(arr[i]%2!=0) odd++;
15     ...}
16     ...printf("Positive number is: %d\n",positive);
17     ...printf("Negative number is: %d\n",negative);
18     ...printf("Even number is: %d\n",even);
19     ...printf("Odd number is: %d\n",odd);
```

You, 2 c

```
19     ...printf("Odd number is: %d\n", odd);  
20     ...return 0;  
21 }
```

Output:

```
5  
4  
6  
2  
8  
6  
7  
Positive number is: 7  
Negative number is: 0  
Even number is: 5  
Odd number is: 2
```

Problem No: 2302056_107

Problem Name: Write a C program that prints ten consecutive odd and even numbers after accepting an integer.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int x;
6      int round=1;
7      printf("Enter the number: ");
8      scanf("%d",&x);
9      printf("Even number is: ");
10     for(int i=x+1; round<=20; i++,round++){
11         if(i%2==0){
12             printf("%d ",i);
13         }
14     }
15     round = 1;
16     printf("\n");
17     printf("Odd number is: ");
18     for(int i=x+1; round<=20; i++,round++){
19         if(i%2!=0){
```

You, 2 days


```

20     .....printf("%d ",i);
21     .....}
22     .....}
23     .....return 0;
24     }

```

Output:

```

Enter the number: 50
Even number is: 52 54 56 58 60 62 64 66 68 70
Odd number is: 51 53 55 57 59 61 63 65 67 69

```

Problem No: 2302056_108

Problem Name: Write a C program that reads two integer values and calculates the sum of all odd numbers between them.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int start,end;
6      printf("Enter the starting number: ");
7      scanf("%d",&start);
8      printf("Enter the ending number: ");
9      scanf("%d",&end);
10     for(int i=start; i<=end; i++){
11         if(i%2!=0){
12             printf("%d ",i);
13         }
14     }
15     return 0;
16 }
```

Output:

```
Enter the starting number: 5
Enter the ending number: 15
5 7 9 11 13 15
```

Problem No: 2302056_109

Problem Name: Write a C program to find and print the square of each even and odd value between 1 and a given number ($4 < n < 101$).

Input:

```
1 #include<stdio.h>
2 #include<math.h>
3
4 int main()
5 {
6     int n;
7     printf("Enter the number: ");
8     scanf("%d",&n);
9     printf("Even number square : \n");
10    for(int i=1; i<=n; i++){
11        if(i%2==0){
12            int power = pow(i,2);
13            printf("%d ^ 2 = %d\n",i,power);
14        }
15    }
16    printf("\n");
17    printf("Odd number square : \n");
18    for(int i=1; i<=n; i++){
19        if(i%2!=0){
```

```

20     .... int power = pow(i,2);
21     .... printf("%d ^ 2 = %d\n",i,power);
22     .... }
23     .... }
24     .... return 0;
25     .... }

```

Output:

```

Enter the number: 15
Even number square :
2 ^ 2 = 4
4 ^ 2 = 16
6 ^ 2 = 36
8 ^ 2 = 64
10 ^ 2 = 100
12 ^ 2 = 144
14 ^ 2 = 196

```

```

Odd number square :
1 ^ 2 = 1
3 ^ 2 = 9
5 ^ 2 = 25
7 ^ 2 = 49
9 ^ 2 = 81
11 ^ 2 = 121
13 ^ 2 = 169
15 ^ 2 = 225

```

Problem No: 2302056_110

Problem Name: 110. Classify a number as positive/negative and odd/even, or print "Zero"

Input:

```
1 #include<stdio.h>
2
3 int main()
4 {
5     int x;
6     printf("Enter the checking number:");
7     scanf("%d",&x);
8     x>0 && x%2==0 ? printf("The number is Positive and Even") : x>0 && x%2!=0 ? printf("The number is Positive and Odd") : x<0 && x%2==0 ? printf("The number is Negative and Even") : printf("The number is Negative and Odd");
9     return 0;
10 }
```

Output:

```
Enter the checking number: 35
The number is Positive and Odd
```

Problem No: 2302056_111

Problem Name: Write a C program that accepts an integer from the user and divides all numbers between 1 and 100. Print those numbers where the remainder value is 3.

Input:

```
1 #include<stdio.h>      You, 8 minutes ago • add
2 int main()
3 {
4     int x;
5     printf("Enter the number: ");
6     scanf("%d",&x);
7     printf("\nRemainder value is 3 after divide all numbers between 1 and 100 by %d:\n", x);
8     for(int i=1; i<=100; i++){
9         if(i%x==3) printf("%d\n",i);
10    }
11    return 0;
12 }
```

Output:

```
Enter the number: 65

Remainder value is 3 after divide all numbers between 1 and 100 by 65:
3
68
```

Problem No: 2302056_112

Problem Name: Write a C program that reads seven integer values from the user and finds the highest value and its position.

Input :

```
1  #include<stdio.h>
2
3  int main()
4  {
5      ...int x[7];
6      ...printf("Enter the 7 number: ");
7      ...for(int i=0; i<7; i++){
8          ...scanf("%d",&x[i]);
9      ...}
10     ...int count=1;
11     ...int maximam = x[0];
12     ...for(int i=0; i<7; i++){
13         ...if(x[i]>maximam){
14             ...maximam = x[i];
15             ...count = i+1;
16         ...}
17     ...}
18     ...printf("Maximum number is: %d\n",maximam);
19     ...printf("Maximum number position is: %d",count);
20     ...return 0;
21 }
```


Output:

```
Enter the 7 number: 15
10
45
30
35
25
40
Maximum number is: 45
Maximum number position is: 3
```

Problem No: 2302056_113

Problem Name: Write a C program to create and print the sequence of the following example.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      ... int a=1;
6      ... int b=100;
7      ... printf("a ... b\n");
8      ... while(b>-1){
9          ... printf("%d ... %d\n",a,b);
10         ... a+=5,b-=10;
11     ... }
12     ... return 0;
13 }
```

Output:

a	b
1	100
6	90
11	80
16	70
21	60
26	50
31	40
36	30
41	20
46	10
51	0

Problem No: 2302056_114

Problem Name: Write a C program that accepts two integer values and calculates the sum of all even values between them.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int start,end;
6      printf("Enter the starting number: ");
7      scanf("%d",&start);
8      printf("Enter the ending number: ");
9      scanf("%d",&end);
10     int sum = 0;
11     for(int i=start; i<=end; i++){
12         if(i%2==0){
13             sum += i;
14         }
15     }
16     printf("The even number sum between the number of %d to %d = %d",start,end,sum);
17     return 0;
18 }

```

Output:

```

Enter the starting number: 35
Enter the ending number: 55
The even number sum between the number of 35 to 55 = 450

```

Problem No: 2302056_115

Prbolem Name: Write a C program that accepts a pair of numbers from the user and prints the sequence from the lowest to the highest number. Also, print the average value of the sequence.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int start,end;
6      printf("Enter the starting number: ");
7      scanf("%d",&start);
8      printf("Enter the ending number: ");
9      scanf("%d",&end);
10     int sum = 0;
11     if (start > end) {
12         int temp = start;
13         start = end;
14         end = temp;
15     }
16     for(int i=start; i<=end; i++){
17         printf("%d\n",i);
18         sum += i;
19     }
20     float avrg = (float)sum/((end-start)+1);
20     float avrg = (float)sum/((end-start)+1);
21     printf("The even number average between the number of %d to %d = %f",start,end,avrg);
22     return 0;
23 }

```

Output:

```
Enter the ending number: 25
14
15
16
17
18
19
20
21
22
23
24
25
The even number average between the number of 14 to 25 = 19.500000
```

Problem No: 2302056_116

Problem Name: Write a C program that accepts a pair of numbers from the user and prints "Ascending order" if the two numbers are in ascending order, otherwise prints, "Descending order".

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int first, second;
6      printf("Enter the First number: ");
7      scanf("%d",&first);
8      printf("Enter the second number: ");
9      scanf("%d",&second);
10     if (first > second) printf("Descending order");
11     else printf("Ascendding order");
12     return 0;
13 }

```

Output:

```

Enter the First number: 65
Enter the second number: 28
Descending order

```

Problem No: 2302056_117

Problem Name: Write a C program that reads two integers and divides the first number by second, print the result of this division with two digits after the decimal point and prints "Division not possible..!" if the division is not possible.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      float x,y;
6      printf("Enter the First number: ");
7      scanf("%f",&x);
8      printf("Enter the second number: ");
9      scanf("%f",&y);
10     if (y==0) printf("Division not possible...!");
11     else printf("Division is %f",x/y);
12     return 0;
13 }

```

Output:

```

Enter the First number: 75
Enter the second number: 8
Division is 9.375000

```

Problem No: 2302056_118

Problem Name: Write a C program that reads five subject marks (0-100) of a student and calculates the average of these marks.


```

1  #include<stdio.h>
2
3  int main()
4  {
5      ... int x;
6      ... float sum = 0;
7      ... printf("Input five subject marks(0-100):\n");
8      ... for(int i=1; i<=5; i++){
9          ... scanf("%d",&x);
10         ... sum+=x;
11     ... }
12     ... printf("Average is: %f",sum/5);
13     ... return 0;
14 }

```

Output:

```

• Input five subject marks(0-100):
88
78
93
89
86
Average is: 86.800003

```

Problem No: 2302056_119

Problem Name: Write a C program to calculate the sum of all numbers between two given numbers (inclusive) not divisible by 7.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int start,end;
6      printf("Enter the First number:");
7      scanf("%d",&start);
8      printf("Enter the second number:");
9      scanf("%d",&end);
10     int sum = 0;
11     if (start > end){
12         int temp = start;
13         start = end;
14         end = temp;
15     }
16     for(int i=start; i<=end; i++){
17         if(i%7!=0) sum+=i;
18     }
19     printf("The sum is between %d to %d (Except multiple number by 7): %d",start,end,sum);
20     return 0;
21 }
```

Output:

```
Enter the First number: 15
Enter the second number: 25
The sum is between 15 to 25 (Except multiple number by 7): 199
```

Problem No: 2302056_120

Problem Name: Write a C program to print a sequence from 1 to a given (integer) number, inserting a comma between these numbers. There will be no comma after the last character.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int n;
6      printf("Enter the number: ");
7      scanf("%d",&n);
8      for(int i=1; i<=n; i++){
9          printf("%d,",i);
10     }
11     return 0;
12 }
```

You, 11 hours ago • add

Output:

```
Enter the number: 25
1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,
```

Problem No: 2302056_121

Problem Name: Write a C program that reads an integer and finds all the divisors of the said integer.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int n;
6      printf("Enter the number: ");
7      scanf("%d",&n);
8      for(int i=1; i<=n; i++){
9          if(n%i==0) printf("%d\n",i);
10     }
11     return 0;
12 }
```

Output:

Enter the number: 36

1
2
3
4
6
9
12
18
36

Problem No: 2302056_122

Problem Name: Write a C program that reads two integers m, n and computes the sum of n even numbers starting from m.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int start,end;
6      printf("Enter the starting number:");
7      scanf("%d",&start);
8      printf("Enter the ending number:");
9      scanf("%d",&end);
10     int sum = 0;
11     for(int i=1,j=start; i<=end; j++){
12         if(j%2==0){
13             sum+=j;
14             i++; //eikhane i dite hobe karon i tokhoni increament hobe jokhon j even number pabe .karon n ta even
                  count korte bolece...jdi i++ for loop () bracket e kori thle seikhtre alltime i++ hobe seta hok even hok
                  odd...kintu amader to n ta even sum caise sejnno jdi for() loop i i++ kori seikhete nta even hobe na half
                  ta hobe karon odd er somoi o i++ hocce sejonno
15         }
16     }
17     printf("\nSum of %d even numbers starting from %d = %d: ",start,end,sum);
18     return 0;
19 }
```

Output:

```
Enter the starting number: 20
Enter the ending number: 60

Sum of 20 even numbers starting from 60 = 4740:
```

Problem No: 2302056_123

Problem Name: Write a C program that reads two integers m, n and computes the sum of n odd numbers starting from m.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int start,end;
6      printf("Enter the starting number: ");
7      scanf("%d",&start);
8      printf("Enter the ending number: ");
9      scanf("%d",&end);
10     int sum = 0;
11     for(int i=1,j=start; i<=end; j++){
12         if(j%2!=0){
13             sum+=j;
14             i++; //eikhane i dite hobe karon i tokhoni increament hobe jokhon j even number pabe .karon n ta even
                  count korte bolece...jdi i++ for loop () bracket e kori thle seikhtre alltime i++ hobe seta hok even hok
                  odd...kintu amader to n ta even sum caise sejnno jdi for() loop i i++ kori seikhte nta even hobe na half
                  ta hobe karon odd er somoi o i++ hocce sejonno
15         }
16     }
17     printf("\nSum of %d even numbers starting from %d = %d ",start,end,sum);
18     return 0;
19 }
```

Output:

```
Enter the starting number: 65
Enter the ending number: 5

Sum of 65 even numbers starting from 5 = 345
```

Problem No: 2302056_124

Problem Name: Write a C program that reads an array of integers (length 7), replaces every negative or null element with 1 and prints the array elements.

Input:

```

1  #include<stdio.h>
2
3  ∨ int main()
4  {
5      ....int arr[7];
6      ....printf("Enter the 7 number: \n");
7  ∨ ....for(int i=0; i<7; i++){
8      ....|....scanf("%d",&arr[i]);
9      ....}
10     ....printf("Array elements:\n");
11  ∨ ....for(int i=0; i<7; i++){
12  ∨ ....|....if(arr[i]<=0){
13      ....|....|....printf("array_num[%d] = 0\n",i);
14  ∨ ....|....} else{
15      ....|....|....printf("array_num[%d] = %d\n",i,arr[i]);
16      ....|....}
17      ....}
18      ....return 0;
19  }

```

Output:


```
Enter the 7 number:
```

```
15
```

```
12
```

```
-7
```

```
25
```

```
0
```

```
27
```

```
53
```

```
Array elements:
```

```
array_num[0] = 15
```

```
array_num[1] = 12
```

```
array_num[2] = 0
```

```
array_num[3] = 25
```

```
array_num[4] = 0
```

```
array_num[5] = 27
```

```
array_num[6] = 53
```

Problem No: 2302056_125

Problem Name: Write a C program that reads an array of integers (length 7), and replaces the first element of the array by a given number and replaces each subsequent position of the array by the double value of the previous.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int a;
6      printf("Enter the number: ");
7      scanf("%d",&a);
8      int b=1;
9      int multiple = a;
10     for(int i=0; i<7; i++){
11         multiple = 5*b;
12         printf("array_num[%d] = %d\n",i,multiple);
13         b*=2;
14     }
15     return 0;
16 }

```

Output:

```

Enter the number: 5
array_num[0] = 5
array_num[1] = 10
array_num[2] = 20
array_num[3] = 40
array_num[4] = 80
array_num[5] = 160
array_num[6] = 320

```

Problem No: 2302056_126

Problem Name: Write a C program that reads an array (length 7) and prints all array positions that store a value less than or equal to 0.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      int arr[7];
6      printf("Enter the 7 number: \n");
7      for(int i=0; i<7; i++){
8          scanf("%d",&arr[i]);
9      }
10     printf("Array positions that store a value less or equal to 0:\n");
11     for(int i=0; i<7; i++){
12         if(arr[i]<=0){
13             printf("arra_nums[%d] = %d\n",i,arr[i]);
14         }
15     }
16     return 0;
17 }
```

Output:

```
Enter the 7 number:
```

```
-30
```

```
50
```

```
00
```

```
40
```

```
-70
```

```
20
```

```
10
```

```
Array positions that store a value less or equal to 0:
```

```
arra_nums[0] = -30
```

```
arra_nums[2] = 0
```

```
arra_nums[4] = -70
```

Problem No: 2302056_127

Problem No: Write a C program that reads an array of integers (length 8), replaces the 1st element with the 8th, the 2nd with the 7th and so on. Print the final array.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int arr[8];
6      printf("Enter the 8 numbers: \n");
7      for(int i=0; i<8; i++){
8          scanf("%d",&arr[i]);
9      }
10     for(int i=7; i>=0; i--){
11         printf("array_numbers[%d] = %d\n",i,arr[i]);
12     }
13     return 0;
14 }

```

Output:

```

Enter the 8 numbers:
25
35
17
-5
29
45
60
65
array_numbers[7] = 65
array_numbers[6] = 60
array_numbers[5] = 45
array_numbers[4] = 29

```

```
array_numbers[3] = -5  
array_numbers[2] = 17  
array_numbers[1] = 35  
array_numbers[0] = 25
```

Problem No: 2302056_129

Problem Name: Write a C program that reads an array (length 10), and replaces the first element of the array by a given number and replaces each subsequent position of the array by one-third the value of the previous.

Input:

```
1  #include<stdio.h>  
2  
3  int main()  
4  {  
5      float a;  
6      printf("Enter the number: ");  
7      scanf("%f",&a);  
8      for(int i=0; i<10; i++){  
9          printf("arr_numbers[%d] = %.4f\n",i,a);  
10         a/=3;  
11     }  
12     return 0;  
13 }
```

Output:

```
Enter the number: 5
arr_numbers[0] = 5.0000
arr_numbers[1] = 1.6667
arr_numbers[2] = 0.5556
arr_numbers[3] = 0.1852
arr_numbers[4] = 0.0617
arr_numbers[5] = 0.0206
arr_numbers[6] = 0.0069
arr_numbers[7] = 0.0023
arr_numbers[8] = 0.0008
arr_numbers[9] = 0.0003
```

Problem No: 2302056_130

Problem Name: Write a C program to create an array of length n and fill the array elements with integer values. Find the smallest value and its position in the array.

Input:

```
1  #include<stdio.h>
2
3  int main()
4  {
5      ....int n;
6      ....printf("Enter the number: ");
7      ....scanf("%d",&n);
8      ....int arr[n];
9      ....for(int i=0; i<n; i++){
10         ....scanf("%d",&arr[i]);
11     ....}
12     ....int position=0;
13     ....int minimum=arr[0];
14     ....for(int i=1; i<n; i++){
15         ....if(arr[i]<minimum){
16             ....minimum=arr[i];
17             ....position=i;
18         ....}
```



```
19     ....}
20     ....printf("The minimum number is : %d\n",minimum);
21     ....printf("The minimum number position is : %d",position);
22     ....return 0;
23 }
```

Output:

```
• Enter the number: 5
35
17
-5
45
36
The minimum number is : -5
The minimum number position is : 2
```

Problem No: 2302056_131

Problem Name: Write a C program that accepts two strings and checks whether the second string is present in the last part of the first string.

Input:

```
1  #include<stdio.h>
2  #include<string.h>
3
4  int main()
5  {
6      ....int n1_lenght,n2_lenght;
7      ....char n1[50],n2[50];
8      ....printf("Enter the first string: ");
9      ....scanf("%s",n1);
10     ....printf("Enter the second string: ");
11     ....scanf("%s",n2);
12     ....n1_lenght=strlen(n1);
13     ....n2_lenght=strlen(n2);
14     ....if(n1_lenght<n2_lenght) printf("Not present.");
15     ....if(n1_lenght==n2_lenght){
16     ....    ....if(strcmp(n1,n2)){
17     ....        ....printf("Present");
18     ....        ....}
```

```

18     ....}
19     ...}
20  ✓ ....if(n1_lenght>n2_lenght){
21  ✓ ....    ....if(strcmp(&n1[n1_lenght-n2_lenght-1],n2)){
22     ....    ....    printf("Present.");
23     ....    ....}
24     ....}
25     ....return 0;
26 }

```

Output:

```

Enter the first string: abcdef
Enter the second string: ef
Present.

```

Problem No: 2302056_132

Problem Name: Write a C program to find the heights of the top three buildings in descending order from eight given buildings.

Input:

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int heights[8];
6      int topThree[3] = {-1, -1, -1};
7      printf("Enter the heights of 8 buildings:\n");
8      for (int i = 0; i < 8; i++) {
9          printf("Height of building %d: ", i + 1);
10         scanf("%d", &heights[i]);
11     }
12     for (int i = 0; i < 8; i++) {
13         if (heights[i] > topThree[0]) {
14             topThree[2] = topThree[1];
15             topThree[1] = topThree[0];
16             topThree[0] = heights[i];
17         } else if (heights[i] > topThree[1]) {
18             topThree[2] = topThree[1];
19             topThree[1] = heights[i];
20         } else if (heights[i] > topThree[2]) {
```

```

21     ....topThree[2] = heights[i];
22     ....}
23     ....}
24     ....printf("The top three building heights in descending order are: %d, %d, %d\n",
25     ....topThree[0], topThree[1], topThree[2]);
26     ....return 0;
27 }
28

```

Output:

```

Enter the heights of 8 buildings:
Height of building 1: 25
Height of building 2: 15
Height of building 3: 45
Height of building 4: 22
Height of building 5: 35
Height of building 6: 18
Height of building 7: 95
Height of building 8: 65
The top three building heights in descending order are: 95, 65, 45

```

Problem No: 2302056_133

Problem Name: Write a C program to calculate the sum of two given integers and count the number of digits in the sum value.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int a,b;
6      printf("Enter 1st number:");
7      scanf("%d",&a);
8      printf("Enter 2nd number:");
9      scanf("%d",&b);
10     int sum=a+b;
11     int count=0;
12     while(sum>0){
13         sum=sum/10;
14         count++;
15     }
16     printf("Number of digits of the sum value of the said numbers: %d",count);
17     return 0;
18 }

```

Output:

```

Enter 1st number: 65
Enter 2nd number: 85
Number of digits of the sum value of the said numbers: 3

```

Problem No: 2302056_134

Problem Name: Write a C program to check whether the three given lengths (integers) of three sides of a triangle form a right triangle or not. Print "Yes" if the given sides form a right triangle otherwise print "No".

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int a,b,c;
6      printf("Enter 1st number:");
7      scanf("%d",&a);
8      printf("Enter 2nd number:");
9      scanf("%d",&b);
10     printf("Enter 3rd number:");
11     scanf("%d",&c);
12     if((a*a)+(b*b)==(c*c) || (a*a)+(c*c)==(b*b) || (b*b)+(c*c)==(a*a)) printf("It is a right angle triangle!");
13     else printf("It is not a right angle triangle!");
14     return 0;
15 }

```

Output:

```

Enter 1st number: 12
Enter 2nd number: 11
Enter 3rd number: 13
It is not a right angle triangle!

```

Problem No: 2302056_135

Problem Name: Write a C program that reads an integer n and finds the number of combinations of a, b, c and d ($0 \leq a, b, c, d \leq 9$) where (a + b + c + d) will be equal to n.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      int x;
6      int count = 0;
7      printf("Enter the number(between 1 to 50): ");
8      scanf("%d",&x);
9      for(int a=0; a<=9; a++){
10         for(int b=0; b<=9; b++){
11             for(int c=0; c<=9; c++){
12                 for(int d=0; d<=9; d++){
13                     if(a+b+c+d==x){
14                         printf("%d + %d + %d + %d\n",a,b,c,d);
15                         count++;
16                     }
17                 }
18             }
19         }
20     }
21     printf("Total number of combinations: %d",count);
22     return 0;
23 }

```

Output:

17 - 155 Verc

• Enter the number(between 1 to 50): 5

$0 + 0 + 0 + 5$

$0 + 0 + 1 + 4$

$0 + 0 + 2 + 3$

$0 + 0 + 3 + 2$

$0 + 0 + 4 + 1$

$0 + 0 + 5 + 0$

$0 + 1 + 0 + 4$

$0 + 1 + 1 + 3$

$0 + 1 + 2 + 2$

$0 + 1 + 3 + 1$

$0 + 1 + 4 + 0$

$0 + 2 + 0 + 3$

$0 + 2 + 1 + 2$

$0 + 2 + 2 + 1$

$0 + 2 + 3 + 0$

$0 + 3 + 0 + 2$

$0 + 3 + 1 + 1$

$0 + 3 + 2 + 0$

$0 + 4 + 0 + 1$

$0 + 4 + 1 + 0$

$0 + 5 + 0 + 0$

$1 + 0 + 0 + 4$

$1 + 0 + 1 + 3$

$1 + 0 + 2 + 2$

$1 + 0 + 3 + 1$

$1 + 0 + 4 + 0$

```
1 + 1 + 0 + 3
1 + 1 + 1 + 2
1 + 1 + 2 + 1
1 + 1 + 3 + 0
1 + 2 + 0 + 2
1 + 2 + 1 + 1
1 + 2 + 2 + 0
1 + 3 + 0 + 1
1 + 3 + 1 + 0
1 + 4 + 0 + 0
2 + 0 + 0 + 3
2 + 0 + 1 + 2
2 + 0 + 2 + 1
2 + 0 + 3 + 0
2 + 1 + 0 + 2
2 + 1 + 1 + 1
2 + 1 + 2 + 0
```

```
2 + 2 + 0 + 1
2 + 2 + 1 + 0
2 + 3 + 0 + 0
3 + 0 + 0 + 2
3 + 0 + 1 + 1
3 + 0 + 2 + 0
3 + 1 + 0 + 1
3 + 1 + 1 + 0
3 + 2 + 0 + 0
4 + 0 + 0 + 1
4 + 0 + 1 + 0
4 + 1 + 0 + 0
5 + 0 + 0 + 0
Total number of combinations: 56
```

Problem No: 2302056_136

Problem Name: Write a C program to find prime numbers that are less than or equal to a given integer.

Input:

```
1  #include <stdio.h>
2  #include <stdbool.h>
3
4  bool checkPrime(int number){
5      ... if(number<=1) return false; ...
6      ... for (int i=2; i*i<=number; i++) {
7          ... if(number%i==0) return false;
8          ... }
9      ... return true;
10 }
11
12 int main()
13 {
14     ... int x;
15     ... int count = 0;
16     ... printf("Enter the number: ");
17     ... scanf("%d",&x);
18     ... for(int i=2; i<=x; i++) {
19         ... if(checkPrime(i)) {
20             ... count++;
21         ... }
22     ... }
23     ... printf("Number of prime numbers which are less than or equal to %d = %d",x,count);
24     ... return 0;
25 }
```

Output:

```
Enter the number: 123
Number of prime numbers which are less than or equal to 123 = 30
```

Problem No: 2302056_137

Problem Name: Write a C program to check if a point (x, y) is within a triangle or not. Three points make up a triangle.

Input:

x1,y1,x2,y2,x3,y3,xp,yp separated by a single space

Input:

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  float calculateArea(float x1, float y1, float x2, float y2, float x3, float y3){
5      ...return abs((x1*(y2-y3) + x2*(y3-y1) + x3*(y1-y2)) / 2.0);
6  }
7
8  int main(){
9      ...float x1, y1, x2, y2, x3, y3, x, y;
10     ...printf("Enter value for(x1,y1): \n");
11     ...scanf("%f%f", &x1, &y1);
12     ...printf("Enter value for(x2,y2): \n");
13     ...scanf("%f%f", &x2, &y2);
14     ...printf("Enter value for(x3,y3): \n");
15     ...scanf("%f%f", &x3, &y3);
16     ...printf("Enter value for(x,y): \n");
17     ...scanf("%f%f", &x, &y);
18     ...float totalArea = calculateArea(x1, y1, x2, y2, x3, y3);
19     ...float area1 = calculateArea(x, y, x2, y2, x3, y3);
20     ...float area2 = calculateArea(x1, y1, x, y, x3, y3);
21     ...float area3 = calculateArea(x1, y1, x2, y2, x, y);
22     ...if (totalArea == (area1 + area2 + area3)){
23         ...printf("The point (%.2f, %.2f) is inside the triangle.\n", x, y);
24     ...} else{
25         ...printf("The point (%.2f, %.2f) is outside the triangle.\n", x, y);
26     ...}
27     ...return 0;
28 }

```

Output:

```
Enter value for(x1,y1):  
5  
2  
Enter value for(x2,y2):  
6  
4  
Enter value for(x3,y3):  
3  
9  
Enter value for(x,y):  
5  
2  
The point (5.00, 2.00) is inside the triangle.
```

Problem No: 2302056_138

Problem Name: Write a C program to test whether two lines are parallel or not. The four points are P(x1, y1), Q(x2, y2), R(x3, y3) and S(x4, y4), check PQ and RS are parallel are not.

Input:

```

1  #include<stdio.h>
2
3  int main()
4  {
5      double x1, y1, x2, y2, x3, y3, x4, y4, slope_PQ, slope_RS;
6      printf("Enter value for P(x1,y1): \n");
7      scanf("%lf%lf",&x1,&y1);
8      printf("Enter value for Q(x2,y2): \n");
9      scanf("%lf%lf",&x2,&y2);
10     printf("Enter value for R(x3,y3): \n");
11     scanf("%lf%lf",&x3,&y3);
12     printf("Enter value for S(x,y): \n");
13     scanf("%lf%lf",&x4,&y4);
14
15     if(x1!=x2) slope_PQ=(y2-y1)/(x2-x1);
16     if(x1!=x2) slope_RS=(y4-y3)/(x4-x3);
17
18     if(slope_PQ==slope_RS) printf("PQ and RS are parallel.");
19     else printf("PQ and RS are not parallel!");
20     return 0;
21 }

```

Output:

```
# 17 1501EXC
Enter value for P(x1,y1):
5
7
Enter value for Q(x2,y2):
3
9
Enter value for R(x3,y3):
8
9
Enter value for S(x,y):
5
6
PQ and RS are not parallel!
```

Problem No: 2302056_142

Problem Name: Write a C program that reads the two adjoining sides and the diagonal of a parallelogram and checks whether the parallelogram is a rectangle or a rhombus.

Input:

```
1  #include<stdio.h>
2  int main() {
3      int lenght, width, diagonal;
4      printf("Enter adjoint sides:");
5      scanf("%d %d", &lenght, &width);
6      printf("Enter the diagonal:");
7      scanf("%d", &diagonal);
8      if (lenght*lenght + width*width == diagonal*diagonal) printf("It is a Rectangle\n");
9      else printf("It is not a Rectangle\n");
10     if (lenght==width) printf("It is a Rhombus\n");
11     else printf("It is not a Rhombus\n");
12     return 0;
13 }
```

Output:


```
Enter adjoint sides: 3
4
Enter the diagonal: 5
It is a Rectangle
It is not a Rhombus
```

Problem No: 2302056_143

Problem Name: Write a C program to find the difference between the largest integer and the smallest integer, which are created by 8 numbers from 0 to 9. The number that can be rearranged shall start with 0 as in 00135668.

Input:

```

1  #include <stdio.h>
2
3  int main() {
4      ....int digits[8], ascending_digit = 0, descending_digit = 0, temp;
5      ....printf("Enter 8 digits (0-9) separated by spaces: ");
6      ....for (int i = 0; i < 8; i++) {
7          ....scanf("%d", &digits[i]);
8          ....if (digits[i] < 0 || digits[i] > 9) {
9              ....printf("Error: All inputs must be digits between 0 and 9.\n");
10             ....return 1;
11         ....}
12     ....}
13     ....for (int i = 0; i < 8; i++) {
14         ....for (int j = i + 1; j < 8; j++) {
15             ....if (digits[i] > digits[j]) {
16                 ....temp = digits[i];
17                 ....digits[i] = digits[j];
18                 ....digits[j] = temp;
19             ....}
20         ....}
21     ....}
22
23     ....for (int i = 0; i < 8; i++) {
24         ....ascending_digit = ascending_digit * 10 + digits[i];
25     ....}
26
27     ....for (int i = 0; i < 8; i++) {
28         ....for (int j = i + 1; j < 8; j++) {
29             ....if (digits[i] < digits[j]) {
30                 ....temp = digits[i];
31                 ....digits[i] = digits[j];
32                 ....digits[j] = temp;
33             ....}
34         ....}
35     ....}

```

```

36     ....for (int i = 0; i < 8; i++) {
37     ....    ....discending_digit = discending_digit * 10 + digits[i];
38     ....}
39     ....int difference = discending_digit - ascending_digit;
40     ....printf("Smallest number: %d\n", ascending_digit);
41     ....printf("Largest number: %d\n", discending_digit);
42     ....printf("Difference: %d\n", difference);
43     ....return 0;
44 }

```

Output:

```

Enter 8 digits (0-9) separated by spaces: 2 4 5 3 8 9 7 6^Z
Smallest number: 23456789
Largest number: 98765432
Difference: 75308643

```

Problem No: 2302056_144

Problem Name: Write a C program to create the maximum number of regions obtained by drawing n given straight lines.

Input:

```

1  #include <stdio.h>
2
3  int main() {
4      ....long n;
5      ....printf("Input number of straight lines:\n");
6      ....scanf("%ld", &n);
7      ....printf("Maximum number of regions obtained by drawing %ld given straight lines:\n", n);
8      ....printf("%ld\n", (n*n+n+2)/2);
9      ....return 0;
10 }

```

Output:

Input number of straight lines:

2

Maximum number of regions obtained by drawing 2 given straight lines:

4

Problem No: 2302056_145

Problem Name: