

Problem 52:

Solution :

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

    int arr[n];
    int min_value = __INT_MAX__;
    int min_index = -1;

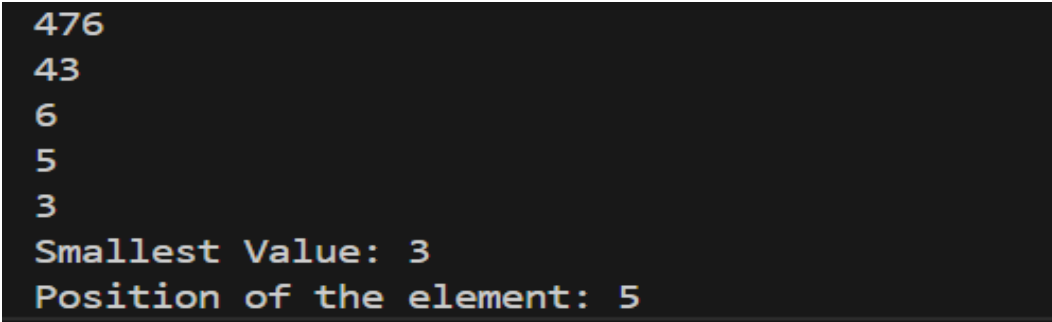
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    for (int i = 0; i < n; i++) {
        if (arr[i] < min_value) {
            min_value = arr[i];
            min_index = i;
        }
    }

    printf("Smallest Value: %d\n", min_value);
    printf("Position of the element: %d\n", min_index + 1);

    return 0;
}
```

Expected Output :



```
476
43
6
5
3
Smallest Value: 3
Position of the element: 5
```

Problem 54:

Solution :

```
#include<stdio.h>
int main() {

    float p, r, t;
    float i;

    printf("Input principle, Rate of interest & time to find simple
interest: ");
    scanf("%f %f %f",&p,&r,&t);

    i=((p*r*t)/100);
    printf("Simple interest = %.4f\n",i);

    return 0;
}
```

Expected Output :

```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc ex.c -o ex } ; if ($?) { .\ex }
Input principle, Rate of interest & time to find simple interest: 100000 10 8
Simple interest = 80000.0000
```

Problem 55:

Solution :

```
#include<stdio.h>
int main(){

    int x, y;
    printf("Input value for x & y: ");
    scanf("%d %d",&x,&y);

    printf("Before swapping the value of x & y: %d %d\n",x,y);
    printf("After swapping the value of x & y: %d %d\n",y,x);

    return 0;
}
```

Expected Output :

```
Input value for x & y: 80 50
Before swapping the value of x & y: 80 50
After swapping the value of x & y: 50 80
```

Problem 57:

Solution :

```
#include<stdio.h>
int main(){

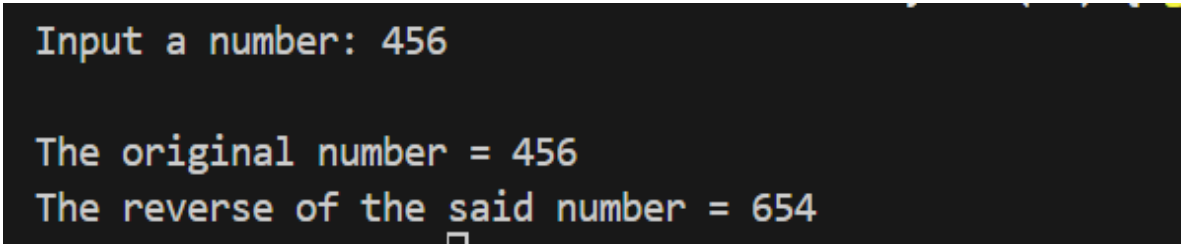
    int n, x, r_n = 0;
    printf("Input a number: ");
    scanf("%d", &n);

    printf("\nThe original number = %d", n);

    while (n >= 1) {
        x = n % 10;
        r_n = r_n * 10 + x;
        n = n / 10;
    }
    printf("\nThe reverse of the said number = %d", r_n);

    return 0;
}
```

Expected Output :



```
Input a number: 456

The original number = 456
The reverse of the said number = 654
```

Problem 58:

Solution : #include <stdio.h>

```
int main() {
```

```
    double a1, a2, a3, a4;
```

```
    double max, min;
```

```
    printf("Input four numbers: \n");
```

```
    scanf("%lf%lf%lf%lf", &a1, &a2, &a3, &a4);
```

```
    if (a1 >= a2 && a1 >= a3 && a1 >= a4) max = a1;
```

```
    else if (a2 >= a1 && a2 >= a3 && a2 >= a4) max = a2;
```

```
    else if (a3 >= a1 && a3 >= a2 && a3 >= a4) max = a3;
```

```
    else max = a4;
```

```
    if (a1 <= a2 && a1 <= a3 && a1 <= a4)
```

```
        min = a1;
```

```
    else if (a2 <= a1 && a2 <= a3 && a2 <= a4)
```

```
        min = a2;
```

```
    else if (a3 <= a1 && a3 <= a2 && a3 <= a4)
```

```
        min = a3;
```

```
    else
```

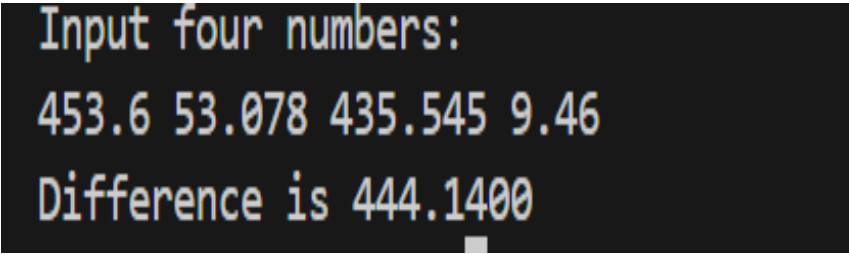
```
        min = a4;
```

```
    printf("Difference is %0.4lf\n", max - min);
```

```
    return 0;
```

```
}
```

Expected Output :



```
Input four numbers:
```

```
453.6 53.078 435.545 9.46
```

```
Difference is 444.1400
```

Problem 59:

Solution :

```
#include<stdio.h>
int main() {
    int num, i, sum = 0;

    printf("Input any number: ");
    scanf("%d", &num);

    printf("1 + ");

    for(i = 2; i <= num - 1; i++)
        printf(" 1/%d +", i);

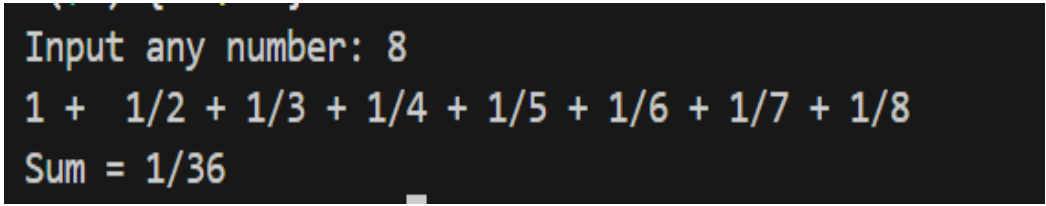
    for(i = 1; i <= num; i++)
        sum = sum + i;

    printf(" 1/%d", num);

    printf("\nSum = 1/%d", sum + 1/num);

    return 0;
}
```

Expected Output :

A screenshot of a terminal window with a black background and white text. The output shows the program's execution for an input of 8. It displays the input number, the series of terms being added (1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6 + 1/7 + 1/8), and the final sum (Sum = 1/36).

```
Input any number: 8
1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6 + 1/7 + 1/8
Sum = 1/36
```

Problem 60:

Solution :

```
#include <stdio.h>
int main() {

    printf("Sun = 0\n");
    printf("Mon = 1\n");
    printf("Tue = 2\n");
    printf("Wed = 3\n");
    printf("Thu = 4\n");
    printf("Fri = 5\n");
    printf("Sat = 6\n");

}
```

Expected Output :

```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?)
Sun = 0
Mon = 1
Tue = 2
Wed = 3
Thu = 4
Fri = 5
Sat = 6
```

Problem 61:

Solution :

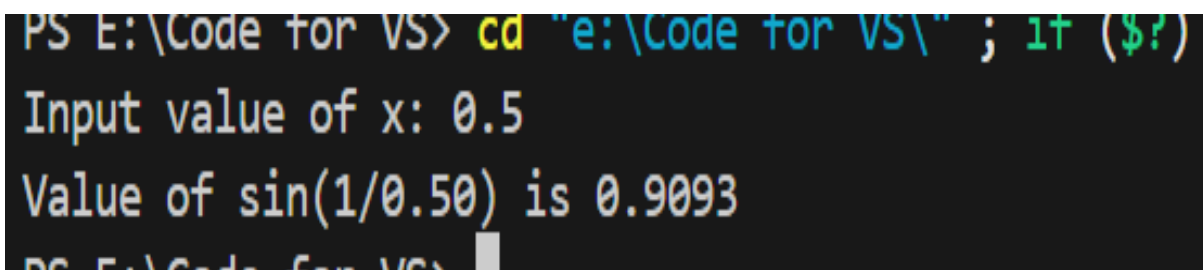
```
#include <stdio.h>
#include <math.h>

int main() {
    double x, result;

    printf("Input value of x: ");
    scanf("%lf", &x);

    if(x==0){
        printf("Enter correct vslue \n");
        return 1;
    }
    result = sin(1/x);
    printf("Value of sin(1/%.2lf) is %.4lf",x,result);
    return 0;
}
```

Expected Output :



```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc 61.c -o 61.exe }
Input value of x: 0.5
Value of sin(1/0.50) is 0.9093
PS E:\Code for VS>
```


Problem 62:

Solution : #include <stdio.h>

```
int main(){
    int n, sum = 0;
    printf("Input a positive number less than 500 : ");
    scanf("%d",&n);

    if(n<1 || n>=500){
        printf("Error, Enter a postive number between 1 to 500
\n");
        return 1;
    }

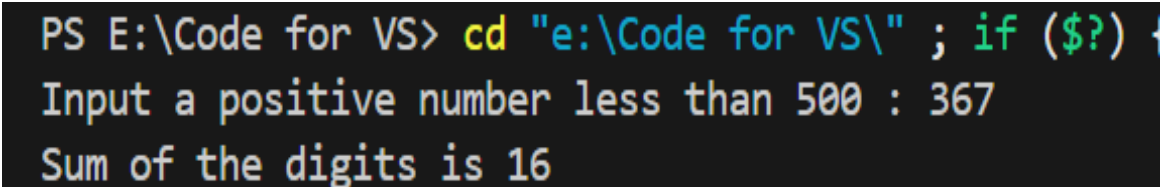
    while(n>0){

        sum+=n%10;
        n/=10;

    }
    printf("Sum of the digits is %d\n", sum);

    return 0;
}
```

Expected Output :



```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) {
Input a positive number less than 500 : 367
Sum of the digits is 16
```

Problem 63:

Solution :

```
#include <stdio.h>
#include <math.h>
int main(){

    int n, j = 1, sum = 0;

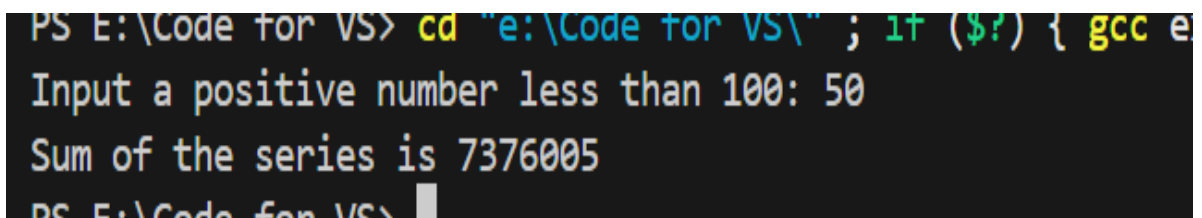
    printf("Input a positive number less than 100: ");
    scanf("%d",&n);

    if(n<1 || n>=100){
        printf("Wrong input\n");
        return 1;
    }

    for (int i=1; j<=n; i++){
        sum+= j*j*j*j;
        j+=i;
    }
    printf("Sum of the series is %d \n",sum);

    return 0;
}
```

Expected Output :



```
PS E:\Code for VS> cd "e:\Code for VS\" ; if ($?) { gcc e
Input a positive number less than 100: 50
Sum of the series is 7376005
PS E:\Code for VS>
```

Problem 65:

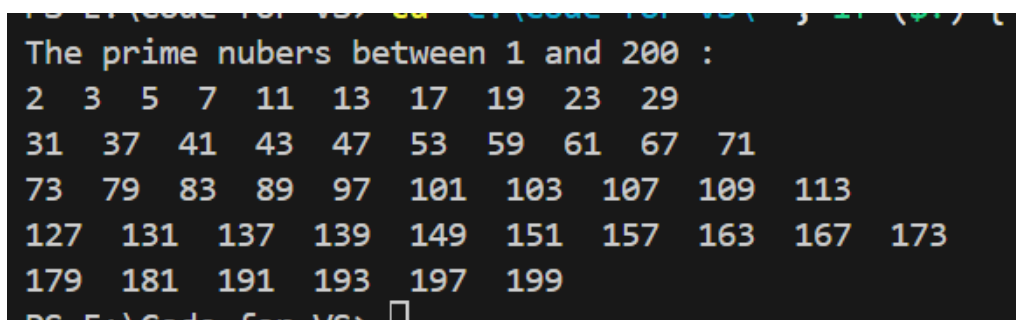
Solution : #include<stdio.h>

```
int main() {
    int flag, count = 0;
    printf("The prime nubers between 1 and 200 : \n");

    for (int i=2; i<=200; i++){
        flag = 1;

        for (int j=2; j <= i/2 && flag==1; j++) {
            if (i%j == 0){
                flag = 0;
            }
        }
        if (flag == 1){
            printf("%d ",i);
            count++;
            if (count % 10 == 0){
                printf("\n");
            }
        }
    }
    printf("\n");
    return 0;
}
```

Expected Output :



```
The prime nubers between 1 and 200 :
2 3 5 7 11 13 17 19 23 29
31 37 41 43 47 53 59 61 67 71
73 79 83 89 97 101 103 107 109 113
127 131 137 139 149 151 157 163 167 173
179 181 191 193 197 199
```

Problem 68:

Solution : #include<stdio.h>

```
int main() {
    long int p;
    int n;
    double q;
    printf("\n=====
    =");
    printf("\n n    2 to power n    2 to power -n");

    printf("\n=====
    =");
    p = 1;

    for (n = 0; n < 11; ++n) {
        if (n == 0)
            p = 1;
        else
            p = p * 2;
        q = 1.0 / (double) p;
        printf("\n%2d    %8d    %20.12lf", n, p, q);
    }
    printf("\n=====
    ");
    return 0;
}
```

Expected Output :

```
=====
n      2 to power n      2 to power -n
=====
0          1      1.000000000000
1          2      0.500000000000
2          4      0.250000000000
3          8      0.125000000000
4         16      0.062500000000
5         32      0.031250000000
6         64      0.015625000000
7        128      0.007812500000
8        256      0.003906250000
9        512      0.001953125000
10       1024      0.000976562500
=====
```

Problem 70:

Solution : `#include <stdio.h>`
`#define N 10`

```
int main(){
    char ch;
    printf("\n");

    for (ch = 65; ch <= 122; ch = ch + 1){
        if (ch > 90 && ch < 97)
            continue;

        printf("[%2d-%c] ",ch,ch);
    }
    return 0;
}
```

Expected Output :

```
[65-A] [66-B] [67-C] [68-D] [69-E] [70-F] [71-G] [72-H] [73-I] [74-J] [75-K] [76-L] [77-M] [78-N] [79-O] [80-P] [81-Q] [82-
R] [83-S] [84-T] [85-U] [86-V] [87-W] [88-X] [89-Y] [90-Z] [97-a] [98-b] [99-c] [100-d] [101-e] [102-f] [103-g] [104-h] [10
5-i] [106-j] [107-k] [108-l] [109-m] [110-n] [111-o] [112-p] [113-q] [114-r] [115-s] [116-t] [117-u] [118-v] [119-w] [120-x
] [121-y] [122-z]
```

Problem 71:

Solution :

```
#include <stdio.h>
int main(){

    char str1[100], str2[100];
    int i;

    printf("Input a string: ");
    scanf("%s", str2);

    for(i=0; str2[i]!='\0'; i++)
        str1[i]=str2[i];
    str1[i]='\0';

    printf("\n");
    printf("Orginal string: %s", str1);
    printf("\nNumber of characters = %d\n", i);

    return 0;
}
```

Expected Output :

```
Input a string: sajabahmed

Orginal string: sajabahmed
Number of characters = 10
```

Problem 72:

Solution : #include <stdio.h>

```
int main(){

    int n;
    printf("Input a value (negative): ");
    scanf("%d",&n);

    printf("Orginal value = %d \n",n);

    if(n<0){
        n = -n;
    }

    printf("Absolute value = %d \n",n);

    return 0;
}
```

Expected Output :

```
Input a value (negative): -56
Orginal value = -56
Absolute value = 56
```

Problem 73:

Solution :

```
#include <stdio.h>
int main() {
    int n1, n2;

    printf("Input the first integer : ");
    scanf("%d",&n1);

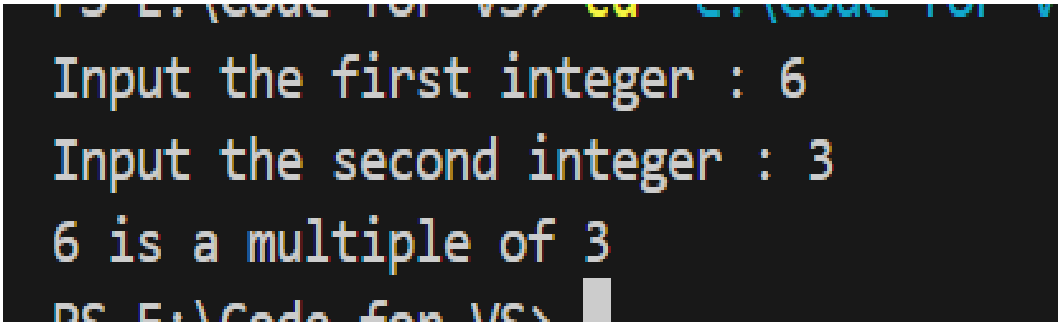
    printf("Input the second integer : ");
    scanf("%d",&n2);

    if(n1%n2==0){
        printf("%d is a multiple of %d \n",n1,n2);
    }

    else{
        printf("%d is not a multiple of %d \n",n1,n2);
    }

    return 0;
}
```

Expected Output :



```
Input the first integer : 6
Input the second integer : 3
6 is a multiple of 3
```

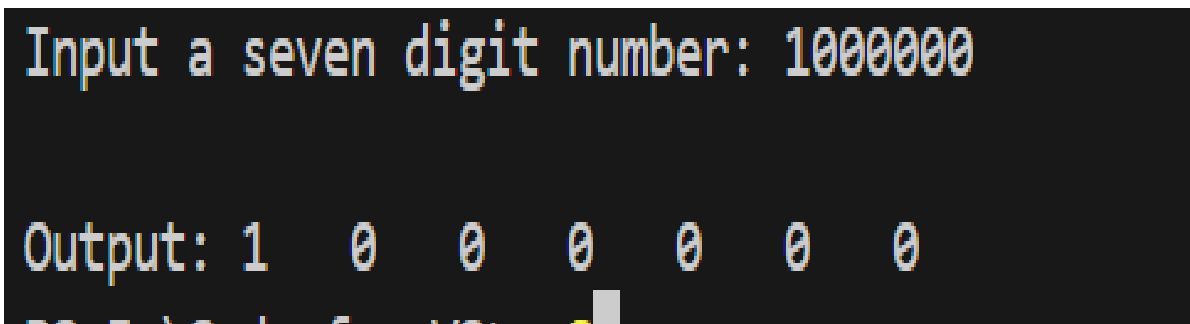

Problem 75:

Solution :

```
#include<stdio.h>
int main()
{
    int n;
    printf( "Input a seven digit number: " );
    scanf("%d", &n);
    printf( "\nOutput: " );
    printf("%d  ", (n/1000000));
    n = n - ((n/1000000)*1000000);
    printf("%d  ", (n/100000));
    n = n - ((n/100000)*100000);
    printf("%d  ", (n/10000));
    n = n - ((n/10000)*10000);
    printf("%d  ", (n/1000));
    n = n - ((n/1000)*1000);
    printf("%d  ", (n/100));
    n = n - ((n/100)*100);
    printf("%d  ", (n/10));
    n = n - ((n/10)*10);
    printf("%d\n", (n%10));

    return 0;
}
```

Expected Output :



```
Input a seven digit number: 1000000

Output: 1  0  0  0  0  0  0
```

Problem 76:

Solution :

```
#include<stdio.h>
int main()
{
    int x;

    printf("Number\tSquare\tCube\n");
    printf("=====\\n");

    for(x=0; x<=20; x++)
        printf("%d\\t%d\\t%d\\n", x, x*x, x*x*x);

    return 0;
}
```

Expected Output :

Number	Square	Cube
=====		
0	0	0
1	1	1
2	4	8
3	9	27
4	16	64
5	25	125
6	36	216
7	49	343
8	64	512
9	81	729
10	100	1000
11	121	1331
12	144	1728
13	169	2197
14	196	2744
15	225	3375
16	256	4096
17	289	4913
18	324	5832
19	361	6859
20	400	8000

Problem 77:

Solution : #include <stdio.h>

```
int main() {
    double p, r, interest;
    int d;
    while (1) {
        printf("Input loan amount (0 to quit): ");
        scanf("%lf", &p);

        if (p == 0) {
            break;
        }

        printf("Input interest rate: ");
        scanf("%lf", &r);

        printf("Input term of the loan in days: ");
        scanf("%d", &d);

        interest = (p * r * d) / 365;
        printf("The interest amount is $%.2f\n", interest);
    }
    return 0;
}
```

Expected Output :

```
Input loan amount (0 to quit): 10000
Input interest rate: 0.1
Input term of the loan in days: 365
The interest amount is $1000.00
```

Problem 78:

Solution : #include<stdio.h>

```
int main()
{
    int x = 10;

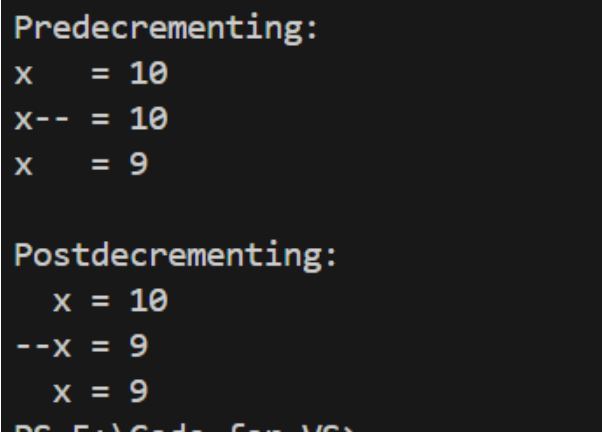
    printf("Predecrementing:\n");
    printf("x  = %d\n", x);
    printf("x-- = %d\n", x--);
    printf("x  = %d\n\n", x);

    x = 10;

    printf("Postdecrementing:\n");
    printf(" x = %d\n", x);
    printf("--x = %d\n", --x);
    printf(" x = %d\n", x);

    return 0;
}
```

Expected Output :



```
Predecrementing:
x  = 10
x-- = 10
x  = 9

Postdecrementing:
 x = 10
--x = 9
 x = 9
```

Problem 79:

Solution :

```
#include<stdio.h>
int main()
{
    int x;

    printf("x\tx+2\tx+4\tx+6\n\n");
    printf("-----\n");

    for(x=1; x<=15; x+=3)
        printf("%d\t%d\t%d\t%d\n", x, (x+2), (x+4), (x+6));

    return 0;
}
```

Expected Output :

x	x+2	x+4	x+6

1	3	5	7
4	6	8	10
7	9	11	13
10	12	14	16
13	15	17	19

Problem 80:

Solution :

```
#include<stdio.h>
int main()
{
```

[illegible]

```

return 0;
}

```

Expected Output :

[illegible]

Problem 81:

Solution : `#include <stdio.h>`

```
int main(){
int size, i, j;
printf("Input the size of the square: ");
scanf("%d", &size);
if (size < 1 || size > 10) {
printf("Size should be in the range 1 to 10\n");
return 0;
}
for (i = 0; i < size; i++)
{
for (j = 0; j < size; j++)
{
if (i == 0 || i == size - 1)
printf("#");
else if (j == 0 || j == size - 1)
printf("#");
else
printf(" ");
}
printf("\n");
}
return 0;
}
```

Expected Output :

[illegible]

Problem 84:

Solution : #include <stdio.h>

```
int main(){
    int n, ctr = 0, sum = 0;
    float avg_value = 0;
    printf("Input the values : ");
    scanf("%d",&n);

    while(n!=888){

        sum+=n;
        ctr++;
        scanf("%d",&n);
    }
    if(ctr){
        avg_value = (float) sum / ctr;
    }
    printf("\nThe average value of the said numbers is %f\n",avg_value);

    return 0;
}
```

Expected Output :

```
Input the values : 10
500
65
888

The average value of the said numbers is 191.666672
```


Problem 86:

Solution : #include <stdio.h>
#include <stdint.h>
#include <limits.h>
#include <float.h>

```
int main() {  
    printf("Size of C data types:\n\n");  
    printf("Type          Bytes\n");  
  
    printf("-----\n");  
  
    printf("char          %lu\n", sizeof(char));  
    printf("int8_t        %lu\n", sizeof(int8_t));  
    printf("unsigned char  %lu\n", sizeof(unsigned char));  
    printf("uint8_t       %lu\n", sizeof(uint8_t));  
    printf("short         %lu\n", sizeof(short));  
    printf("int16_t       %lu\n", sizeof(int16_t));  
    printf("uint16_t      %lu\n", sizeof(uint16_t));  
    printf("int           %lu\n", sizeof(int));  
    printf("unsigned      %lu\n", sizeof(unsigned));  
    printf("long          %lu\n", sizeof(long));  
    printf("unsigned long %lu\n", sizeof(unsigned long));  
    printf("int32_t       %lu\n", sizeof(int32_t));  
    printf("uint32_t      %lu\n", sizeof(uint32_t));  
    printf("long long     %lu\n", sizeof(long long));  
    printf("int64_t       %lu\n", sizeof(int64_t));  
    printf("unsigned long long %lu\n", sizeof(unsigned long long));  
    printf("uint64_t      %lu\n", sizeof(uint64_t));  
    printf("float         %lu\n", sizeof(float));  
    printf("double        %lu\n", sizeof(double));  
    printf("long double   %lu\n", sizeof(long double));  
    printf("bool          %lu\n", sizeof(_Bool));  
  
    return 0;  
}
```

Expected Output :

Type	Bytes

char	1
int8_t	1
unsigned char	1
uint8_t	1
short	2
int16_t	2
uint16_t	2
int	4
unsigned	4
long	4
unsigned long	4
int32_t	4
uint32_t	4
long long	8
int64_t	8
unsigned long long	8
uint64_t	8
float	4
double	8
long double	16
bool	1

Problem 89:

Solution :

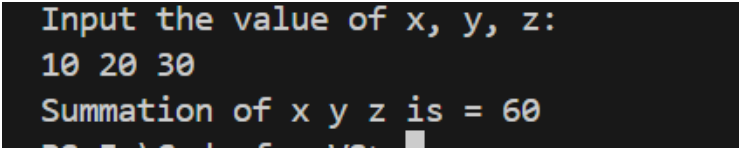
```
#include<stdio.h>
int main(){

    long long x, y, z;
    printf("Input the value of x, y, z: \n");
    scanf("%lld %lld %lld",&x,&y,&z);

    printf("Summation of x y z is = %lld \n", x + y + z);

    return 0;
}
```

Expected Output :

A screenshot of a terminal window showing the output of the C program. The text is as follows:

```
Input the value of x, y, z:
10 20 30
Summation of x y z is = 60
```

Problem 94:

Solution : #include <stdio.h>

```
int main() {
    float weight, height, bmi;
    printf("Input the weight (in kg): ");
    scanf("%f", &weight);

    printf("Input the height (in meters): ");
    scanf("%f", &height);
    bmi = weight / (height * height);

    printf("BMI = %.6f\n", bmi);

    if (bmi < 18.5) {
        printf("Grade: Underweight\n");
    } else if (bmi >= 18.5 && bmi < 24.9) {
        printf("Grade: Normal weight\n");
    } else if (bmi >= 25 && bmi < 29.9) {
        printf("Grade: Overweight\n");
    } else {
        printf("Grade: Obeseity\n");
    }

    return 0;
}
```

Expected Output :

```
Input the weight (in kg): 43
Input the height (in meters): 1.4
BMI = 21.938776
Grade: Normal weight
```