Assignment 6: Looping Statement

Name: Harshal Atre Balkrishna

Roll NO. : A-76

Q1:Print the output for the following series:

a) 1 + 4 – 9 + 16 – 25 + 36 .... + n^2

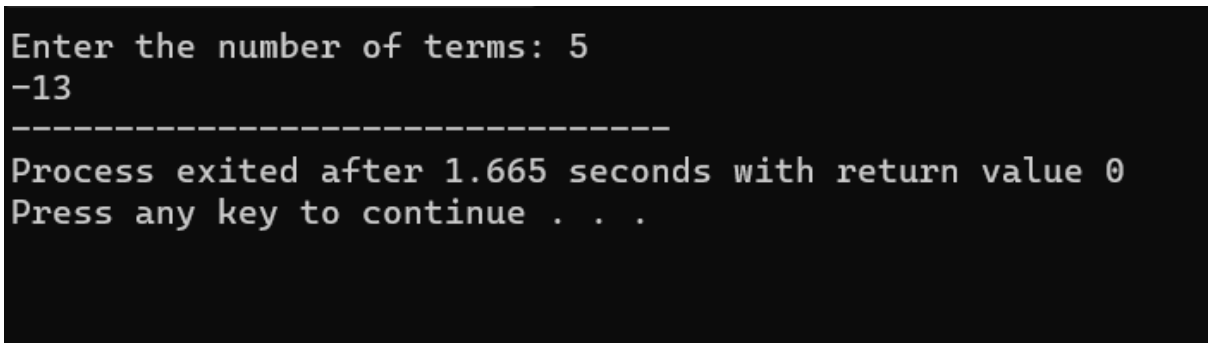
Ans:

Source code:

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

	printf("Enter the number of terms: ");
	scanf("%d",&n);

	for (i=2;i<=n;i++)
	{
		if (i%2==0)
			sum += i*i;
		else
			sum -= i*i;
	}
	sum+=1;
	printf("%d",sum);

	return 0;
```

}

Output:



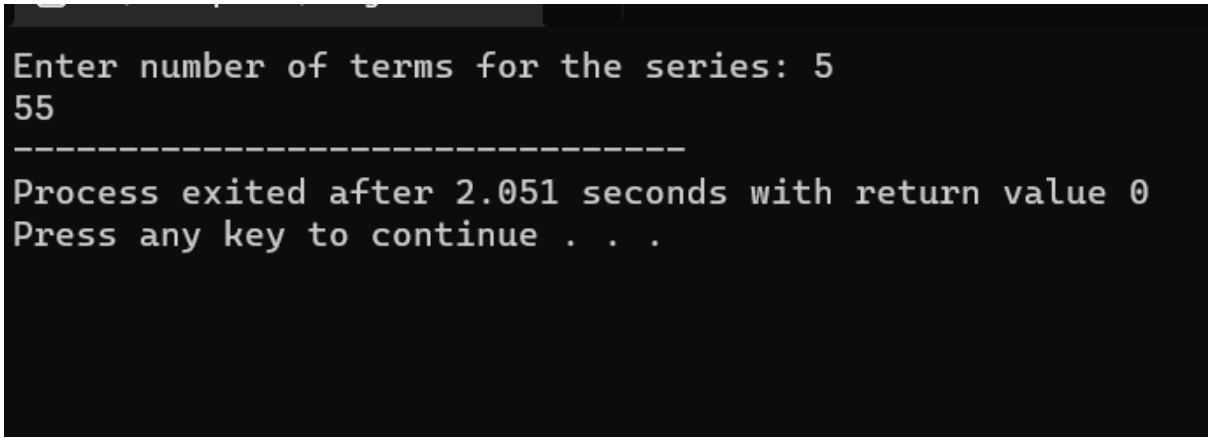
b) 1^2 + 2^2 + 3^2 + 4^2 + ... +n^2

Ans:

Source code:

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

	printf("Enter number of terms for the series: ");
	scanf("%d",&n);

	for(i = 1;i<=n;i++)
	{
		sqsum += i*i;
	}

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

Output:



c) x – x^3/3! + x^5/5! – x^7/7! + x^9/9! ....

Ans:

Source code:

```
#include <stdio.h>
#include <math.h>
int main()
{
	int n,i,j,k,l,x;
	int fact1=1;
	int fact3=1;
	float sum1,sum3,sum;


	printf("Enter number of terms for the series: ");
	scanf("%d",&n);

	printf("Enter value of x: ");
	scanf("%d",&x);

	for(i=1;i<=(2*n-1);i+=4)
	{
		for(k=1;k<=i;k++)
		{
			fact1*=k;
		}
		sum1+=pow(x,i)/fact1;
```

```
	}

	for(j=3;j<=(2*n-1);j+=4)
	{
		for(l=1;l<=j;l++)
		{
			fact3*=l;
		}
		sum3+=pow(x,j)/fact3;
	}

	sum = sum1-sum3;
	printf("%f",sum);

	return 0;
}
```

Output:

```
Enter number of terms for the series: 5
Enter value of x: 6
25.774292
--------------------------------
Process exited after 3.782 seconds with return value 0
Press any key to continue . . .
```

d) Given Number is Armstrong number or Not.

Ans:

Source code:

```
#include <stdio.h>
#include <math.h>
int main()
{
	int dig,n,check_n=0,i=0;
```

```
	printf("Enter a number: ");
	scanf("%d",&n);
	int temp_n=n;

	while(temp_n!=0)
	{
		temp_n /= 10;
		i++;
	}
	temp_n = n;

	while(temp_n!=0)
	{
		dig = temp_n%10;
		check_n += pow(dig,i);
		temp_n/=10;
	}

	if(check_n==n)
		printf("Yes it is an angstrom number");
	else
		printf("No it is not an angstrom number");
	return 0;
}
```

Output:

```
Enter a number: 153
Yes it is an angstrom number
--------------------------------
Process exited after 1.242 seconds with return value 0
Press any key to continue . . .
```

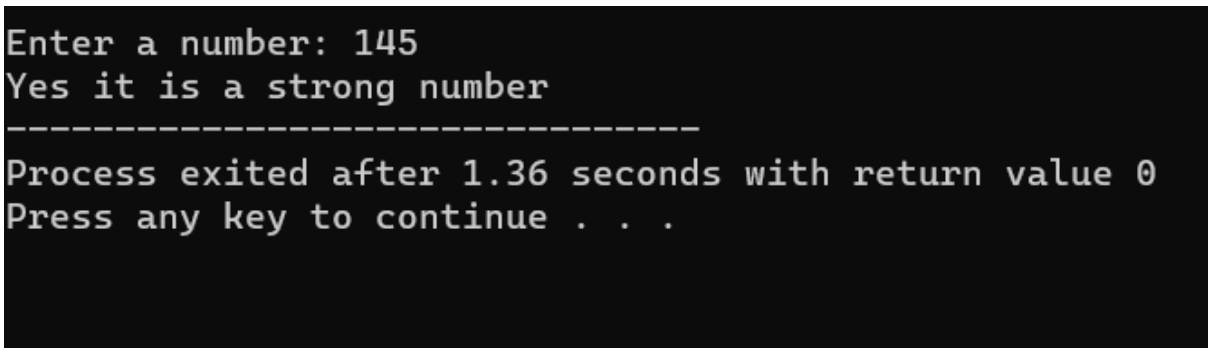
e) Given Number is Strong number or Not.

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int input,temp,digit,check=0,fact=1;
	printf("Enter a number: ");
	scanf("%d",&input);
	temp = input;

	while(temp>0)
	{
		digit = temp%10;
		while(digit>0)
		{
			fact = fact*digit;
			digit--;
		}
		check=check+fact;
		temp=temp/10;
		fact=1;
	}
	if(check==input)
		printf("Yes it is a strong number");
	else
		printf("No it is not a strong number");
	return 0;
}
```

Output:



Q2: Print the following patterns:

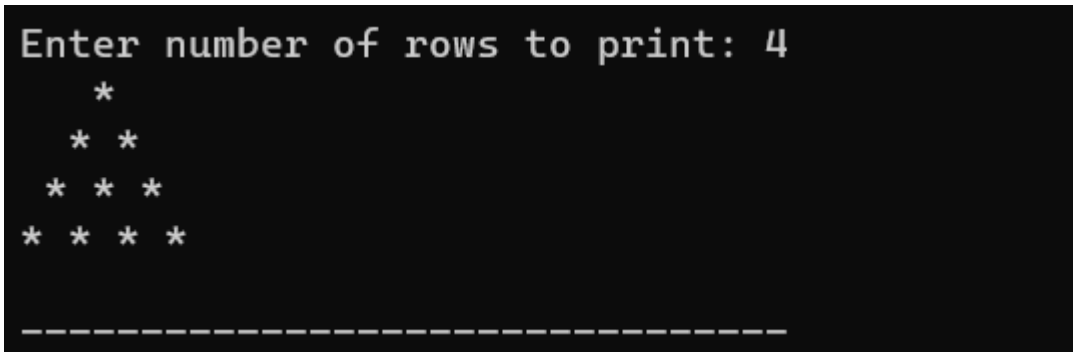
A)

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int n,i,j;

	printf("Enter number of rows to print: ");
	scanf("%d",&n);

	for(i=1;i<=n;i++)
	{
		for(j=1;j<=n-i;j++)
			printf(" ");
		for(j=1;j<=i;j++)
			printf("* ");

		printf("\n");
	}
	return 0;
}
```

Output:



B)

12345
1234
123
12
1

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int i,j,n;
	printf("Enter number of rows: ");
	scanf("%d",&n);
	for(i=n;i>0;i--)
	{
		for(j=1;j<=i;j++)
		{
			printf("%d",j);
		}
		printf("\n");
	}
	return 0;
```

}

Output:

```
Enter number of rows: 5
12345
1234
123
12
1

--------------------------------
```

C)

```
1
4 1
9 4 1
16 9 4 1
25 16 9 4 1
```

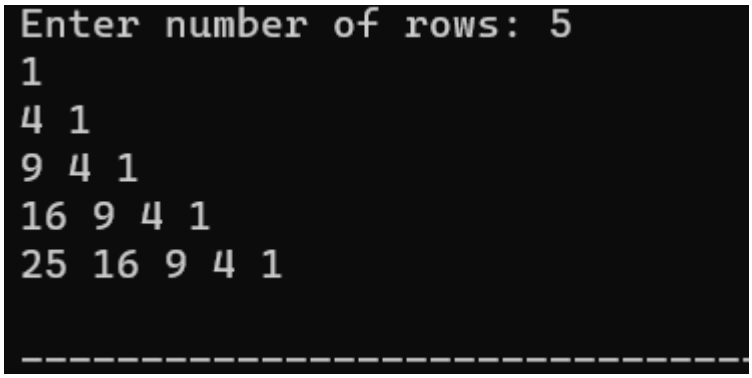
Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int i,j,n;
	printf("Enter number of rows: ");
	scanf("%d",&n);
	for(i=1;i<=n;i++)
	{
		for(j=i;j>=1;j--)
		{
			printf("%d ",j*j);
		}
```

```
            printf("\n");
        }
        return 0;
}
```

Output:



E)

```
    1        1
   12       21
  123      321
 1234     4321
12345    54321
```

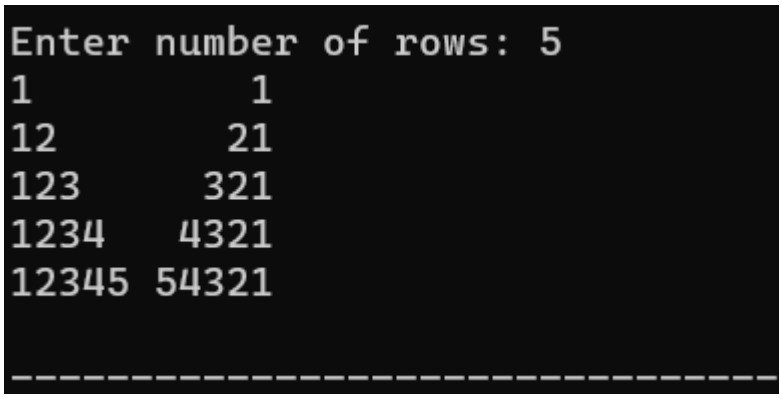
Ans:

Source code:

```
#include <stdio.h>
int main()
{
        int i,j,k,l,n;
        printf("Enter number of rows: ");
        scanf("%d",&n);
        for(i=1;i<=n;i++)
```

```
	{
		for(j=1;j<=i;j++)
		{
			printf("%d",j);
		}
		for(k=1;k<=2*(n-i)+1;k++)
		{
			printf(" ");
		}
		for(l=i;l>=1;l--)
		{
			printf("%d",l);
		}
		printf("\n");
	}
	return 0;
}
```

Output:



F)

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int i,j,k,l,n;
	printf("Enter max number: ");
	scanf("%d",&n);
	for(i=1;i<=n;i++)
	{
		for(j=1;j<=i;j++)
		{
			printf("%d",j);
		}
		printf("\n");
	}
	for(k=n-1;k>=1;k--)
	{
		for(l=1;l<=k;l++)
		{
			printf("%d",l);
		}
```

```
            printf("\n");
        }
        return 0;
}
```

Output:

```
Enter max number: 5
1
12
123
1234
12345
1234
123
12
1
```

F)

```
        A
      A C E
    A C E G I
  A C E G I K M
A C E G I K M O Q
  A C E G I K M
    A C E G I
      A C E
        A
```

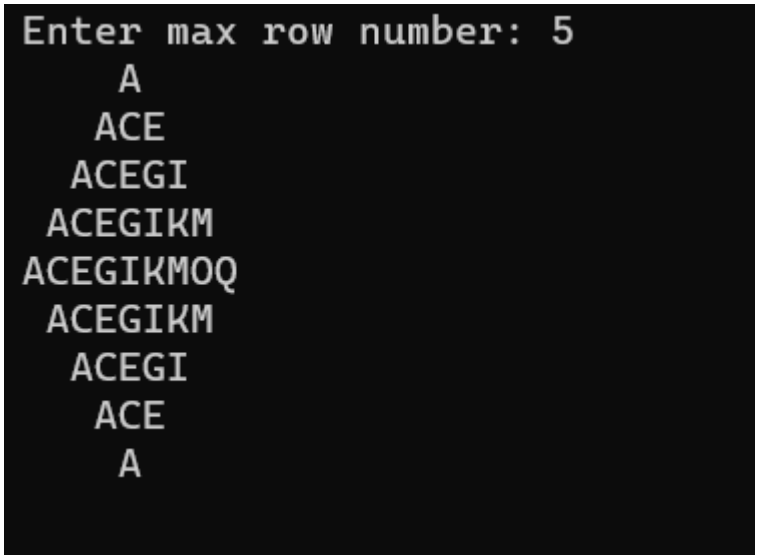
Ans:

Source code:

```
#include <stdio.h>
int main()
{
        int i,j,k,l,m,o,n,c=64;
```

```
	printf("Enter max row number: ");
	scanf("%d",&o);

	for(i=1;i<=o;i++)
	{
		for(j=1;j<=o-i;j++)
		{
			printf(" ");
		}
		for(k=1;k<=4*i-2;k+=2)
		{
			printf("%c",c+k);
		}
		printf("\n");
	}

	for(l=o-1;l>=1;l--)
	{
		for(m=1;m<=o-l;m++)
		{
			printf(" ");
		}
		for(n=1;n<=4*l-2;n+=2)
		{
			printf("%c",c+n);
		}
		printf("\n");
	}
	return 0;
}
```

Output:



Q3: Write a program to do the following task

a) Accept any 2 positive numbers, say n1 and n2. Assume n1 > n2.

b) Print all even numbers that lie between n1 and n2.

c) Print the total number of even numbers between n1 and n2.

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int n1,n2,i,j=0;
	printf("Enter n1 and n2 such that n1>n2: ");
	scanf("%d %d",&n1,&n2);
	for(i=n1-1;i>n2;i--)
	{
		if(i%2==0)
		{
			printf("%d ",i);
			j++;
		}
	}
	printf("\nNo of even numbers between n1 and n2: %d",j);
	return 0;
}
```

Output:

```
Enter n1 and n2 such that n1>n2: 20
10
18 16 14 12
No of even numbers between n1 and n2: 4
```

Q4. Write a program to calculate the sum of the square of each digit of the given number.

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int n,digit,sum=0;
	printf("Enter a number: ");
	scanf("%d",&n);

	while(n>0)
	{
		digit=n%10;
		sum += digit*digit;
		n /= 10;
	}
	printf("%d",sum);
	return 0;
}
```

Output:

```
Enter a number: 4534
66
```

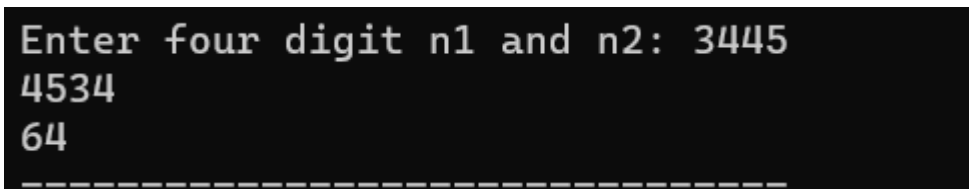
Q5. Accept 2 four-digit positive integers then calculate and display the sum of the product of each pair of digits occupying the same position in the two numbers.

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int n1,n2,dig1,dig2,sum=0;
	printf("Enter four digit n1 and n2: ");
	scanf("%d %d",&n1,&n2);

	while(n1>0)
	{
		dig1=n1%10;
		dig2=n2%10;
		sum+= dig1*dig2;
		n1/=10;
		n2/=10;
	}
	printf("%d",sum);
	return 0;
}
```

Output:



Q6. Given two integers L and R where L ≤ R, the task is to find an integer K such that:

L ≤ K ≤ R.

All the digits of K are distinct.

The value of the expression (L – K) * (K – R) is maximum.

If multiple answers exist then choose the larger value for K.

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int l,r,k,i,temp,temp_k,exp,digit,maxexp=-65536,isDistinct=1;

	printf("Enter value of L: ");
	scanf("%d",&l);
	printf("Enter value of R: ");
	scanf("%d",&r);
	i=l;

	for(k=l;k<=r;k++)
	{
		temp_k=k;
		//Checking whether number is of distinct digits
		while(temp_k>0)
		{
			digit = temp_k%10;
			temp_k/=10;
			temp=temp_k;
			while(temp>0)
			{
				if(temp%10==digit)
				{
					isDistinct = 0;
					break;
				}
				temp/=10;
			}
```

```
                if(isDistinct==0)
                        break;
        }
        // Checking whether expression is max and thereby storing it's value
and k
        if(isDistinct==1)
        {
                exp=(l-k)*(k-r);
                if(exp>maxexp)
                {
                        i=k;
                        maxexp=exp;
                }
                // Checking for same values of maxexp and selecting larger k
                else if (exp==maxexp)
                {
                        if(k>i)
                                i=k;
                }
        }
    }
    printf("Max value of expression is %d for k %d",maxexp,i);
    return 0;
}
```

Output:

```
Enter value of L: 50
Enter value of R: 80
Max value of expression is 104 for k 54
--------------------------------
Process exited after 7.251 seconds with return value 0
Press any key to continue . . .
```

Q7. Count of triples (A, B, C) where A*C is greater than B*B

Given three integers A, B and C. The task is to count the number of triples (a, b, c) such that a*c > b2, where 0 < a <= A, 0 < b <= B and 0 < c <= C.

Ans:

Source code:

```
#include <stdio.h>
int main()
{
	int A,B,C,a,b,c,i;
	printf("Enter triplet A,B,C: ");
	scanf("%d %d %d",&A,&B,&C);

	for(a=1;a<=A;a++)
	{
		for(b=1;b<=B;b++)
		{
			for(c=1;c<=C;c++)
			{
				if((a*c)>(b*b))
					i++;
			}
		}
	}
	printf("%d",i);
	return 0;
}
```

Output:

```
Enter triplet A,B,C: 3 3 3
11
--------------------------------
```