}

Sample Input:-

Enter the number 4.

Sample output

4 is an even number

2)

Sample input:-

Enter the positive number 7

Sample output :

Sum =8

3) Sample input:

Enter the value of n: 6

Sample output:

Sum of even numbers 2 to 6 is: 12

4) Sample input: 1 2 3 4 5 6

Sample output

6 5 4 3 2 1

5) Sample input:

n=3

Sample output

3 is a palindrome

6) Sample input:

Enter a number = 4

Sample output:

4 is an armstrong number

8) Sample input :

Enter a number 3

Sample output

factorial of 3 is 6

9) Sample input: 6

Sample output : 0,1,2,3,4,5

10) Sample input : 8

Sample output :- 0,0,1,1,2,3,5,8,13

1) Sample input:

Enter the number of elements : 4

Enter 4 elements

2 6 5 4

Sample output:

Sum of elements : 17

2) Sample input:

{1, 3, 5, 7} - {2, 4, 6, 8}

Sample output:

1, 2, 3, 4, 5, 6, 7, 8

3) Sample input:

1, 2 3 5 6

Sample output:

array after insertion : 1 2 3 4 5 6

array after deletion 1 82 4 5 6

4) Sample input

2 3 4 5 6 7

Sample output:

7 6 5 4 3 2

5) Sample input:

AMMA

Sample output:

The string is a palindrome.

6) Sample input:

FINNY PAUL

Sample output:

character to Search 'p'

Character 'p' found at position : 5

7) Sample input:

JESUS

Sample output:

( vowels in the string : 2

8)

```
No of rows and Columns of matrix A:2

2

No of rows and Columns of matrix B:2

2

Enter the elements of matrix A

7

8

5

2

Enter the elements of matrix B

1

2

3

4

matrix A;
                              matrix B;
7     8
                              1     2
5     2
                              3     4

matrix multiplication

31      46

11      18
```

Single linked list:

Sample input: Insert end(& head, 3)

Insert Beginning (& head, 1)

Insert End(& head, 5)

Insert middle (head→next, 2)

Insert middle(head→next→next, 4)

Delete node(3, 1)

Sample output:-

Original list: 1→3→2→4→5→Null

List after deletion: 2→4→5→Null

2) Sample input:-

push(& stack, 10);

push(& Stack, 20)

push(& stack, 30)

Sample output:

After pushing

Stack: 10 20 30

popped Elements: 30, 20

After popping:

Stack: 10

3) engue, deque, display

Sample input: Push front (&ds,3)

push front (&ds,2)

push rear (&ds,4)

Push rear (&ds,5)

Display (&ds)

Sample output:

Elements in the ds: 2 3 4 5

popped Element from front : 2

popped Element from front : 3

Elements in the ds : 4,5

4) Infix, postfix

Sample input:

Enter an infix Expression: (a+b)/c-(d*e)

Stack underflow

stack under flow

Output:-

postfix expression (+ab/c-e*)