1. Write a program to perform Push, Pop, and Peek operations on a stack.
2. Write a program to perform Push, Pop, and Peek operations on a stack using linked list.
3. Write a program to reverse a list of given numbers.
4. Write a program to check nesting of parentheses using a stack.
5. Write a program to convert an infix expression into its equivalent postfix notation.
6. Write a program to evaluate a postfix expression.
7. Write a program to convert an infix expression to a prefix expression.
8. Write a program to evaluate a prefix expression.
9. Write a program to calculate the factorial of a given number.
10. Write a program to calculate the GCD of two numbers using recursive functions.
11. Write a program to calculate exp(x,y) using recursive functions.
12. Write a program to print the Fibonacci series using recursion.
13. Write a program to convert the expression “a+b” into “ab+”.
14. Write a program to convert the expression “a+b” into “+ab”.
15. Write a program to implement a stack that stores names of students in the class.
16. Write a program to input two stacks and compare their contents.
17. Write a program to compute F(x, y), where F(x, y) = F(x-y, y) + 1 if y<x And F(x, y) = 0 if x<y.
18. Write a program to compute F(n, r) where F(n, r) can be recursively defined as: F(n, r) = F(n–1, r) + F(n–1, r–1).
19. Write a program to compute Lambda(n) for all positive values of n where Lambda(n) can be recursively defined as: Lambda(n) = Lambda(n/2) + 1 if n>1 and Lambda(n) = 0 if n =1.
20. Write a program to compute F(M, N) where F(M, N) can be recursively defined as: F(M,N) = 1 if M=0 or M≥N≥1 and F(M,N) = F(M–1,N) + F(M–1, N–1), otherwise.
21. Write a program to reverse a string using recursion.