

40 Find the frequency of each character in a string

Algorithm

Start

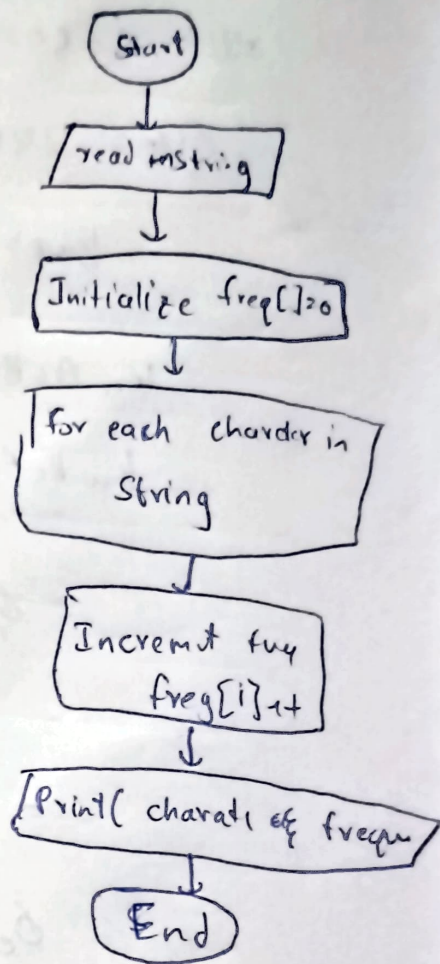
Step-1 Input a string

Step-2 Initialize an array $\text{freq}[26]$ to 0.

Step-3 For each character c in the string:
Increment $\text{freq}[c]$.

Step-4 Print characters with their frequency

Step-5 end.



41. factorial using recursion.

Step-1 start

Step-2 Define a recursive function $\text{fact}(n)$

Step-3 If $n == 0$ (or $n == 1$) return 1

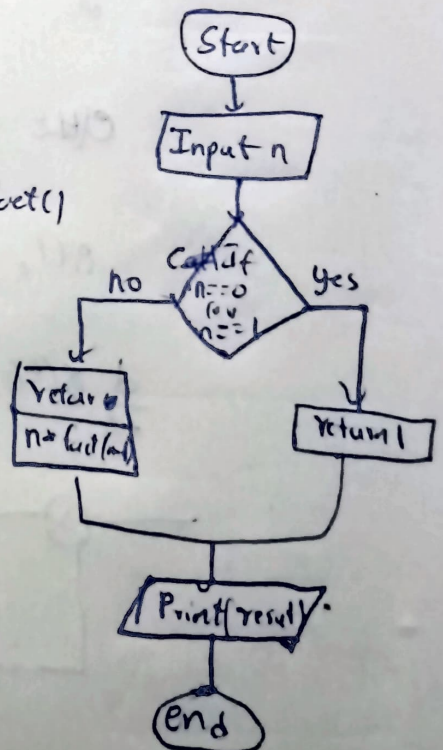
Step-4 else return $n * \text{fact}(n-1)$

Step-5 Input n .

Step-6 Call $\text{fact}(n)$

Step-7 Print result

Step-8 end.



42. Fibonacci series using recursion

Algorithm

Step-1 Start

Step-2 Define function fib(n)

Step-3 If $n == 0 \rightarrow$ return 0

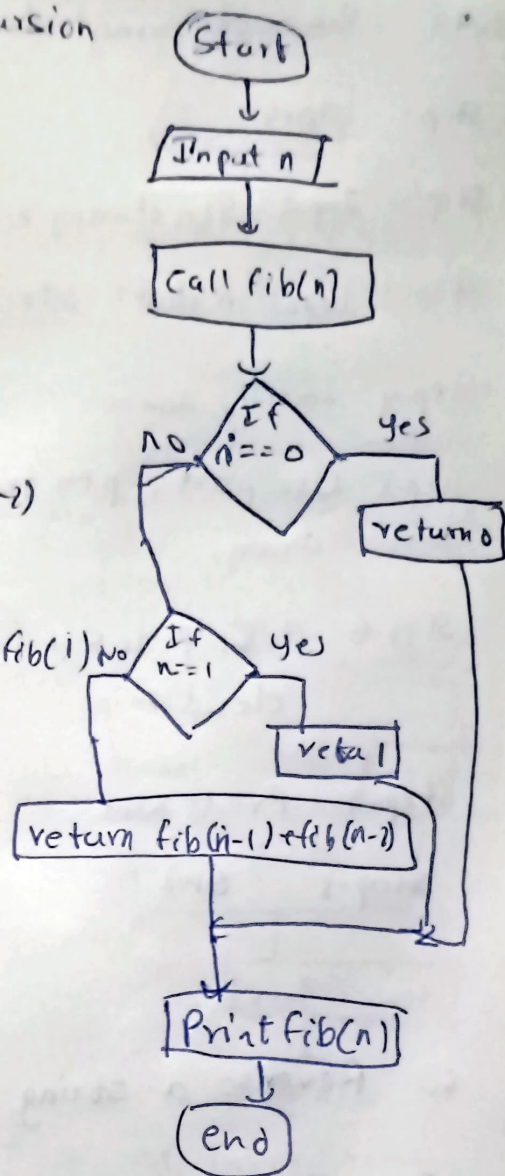
Step-4 If $n == 1 \rightarrow$ return 1

Step-5 else return $\text{fib}(n-1) + \text{fib}(n-2)$

Step-6 Input number of terms n

Step-7 loop from 0 to n-1 print fib(i)

Step-8 end



43. Swap two numbers using pointers.

Step-1 Start

Step-2 Input two numbers a and b.

Step-3 use pointers *p and *q to store address of a and b

Step-4 Swap using temp.

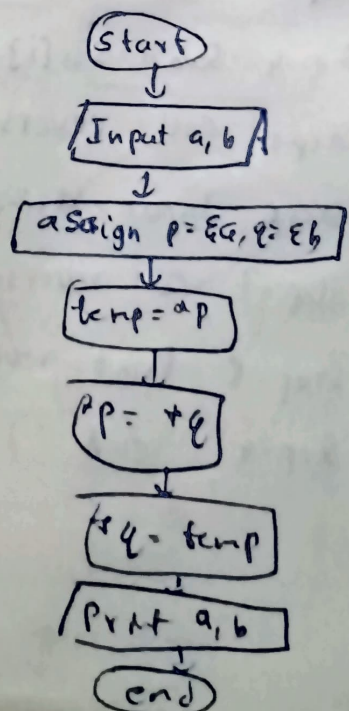
Step-5. $\text{temp} = *p$

Step-6 $*p = *q$

Step-7 $*q = \text{temp}$

Step-8 Print swapped numbers

Step-9 end



Q2 Sum of array elements using pointers

Step 1 Start

Step 2 Input size of array n

Step 3 Input n elements into array

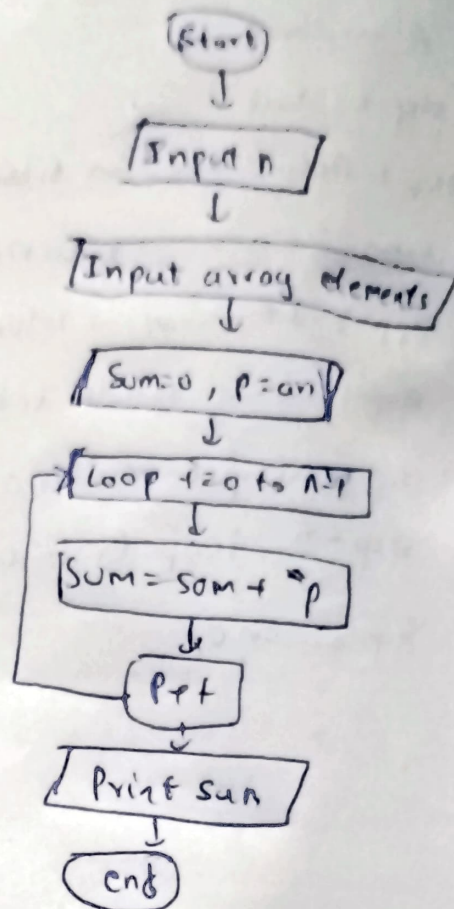
Step 4 Initialize sum

Step 5 Use pointer p to traverse array

Step 6 Add *p to sum for each element

Step 7 Print sum

Step 8 end



Q3 Reverse a string using recursion.

Step 1 Start

Step 2 Define function reverse(str, i, n)

Step 3 If $i \geq n/2$ return

Step 4 Swap $str[i]$ and $str[n-i-1]$

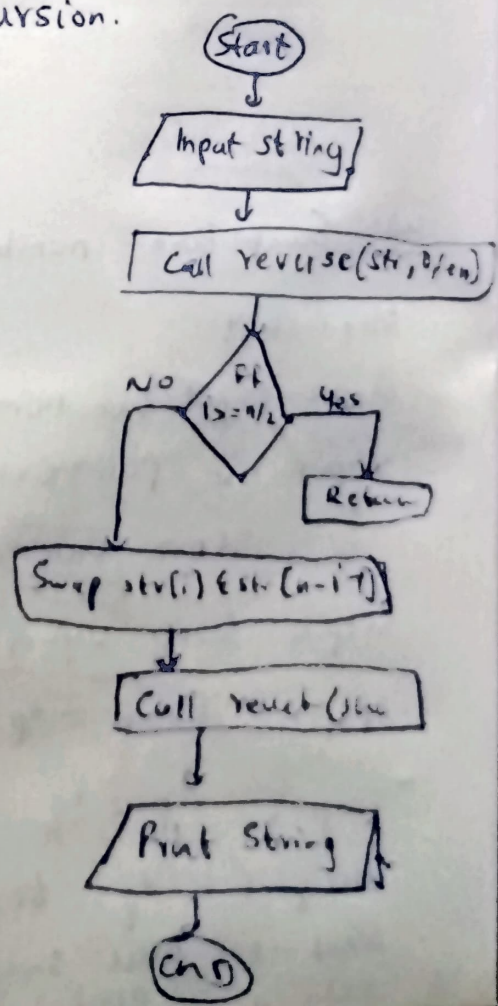
Step 5 Call reverse(str, i+1, n)

Step 6 Input String

Step 7 Call reverse(str, 0, length)

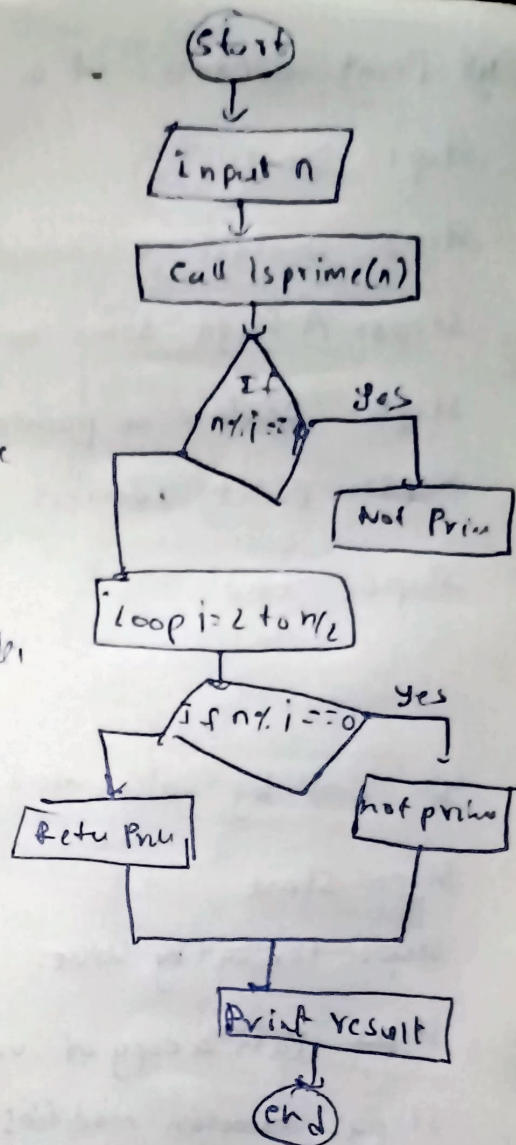
Step 8 Print reversed String

Step 9 end



46 Function to check prime

- Step 1: Start
Step 2: Define function is prime(n)
Step 3: If $n < 1$ return false
Step 4: for $i = 2$ to $n/2$
Step 5: if $n \% i == 0$ return false
Step 6: else return true.
Step 7: Input n.
Step 8: Call is prime(n) and print result
Step 9: end



47 Function to return maximum of three numbers

- Step 1: Start
Step 2: Input a, b, c.
Step 3: If $a > b$ and $a > c$ → $max = a$.
Step 4: Else if $b > c$ → $max = b$.
Step 5: Print max.
Step 6: End.

