Set 1

Basic

1. **User Input and Replace String Template “Hello <<UserName>>, How are you?”**

    a. **I/P** -> Take User Name as Input. *Ensure UserName has min 3 char*
    b. **Logic** -> Replace <<UserName>> with the proper name
    c. **O/P** -> Print the String with User Name

2. **Simulate Stopwatch Program**

    a. Desc -> Write a Stopwatch Program for measuring the time that elapses between the start and end clicks
    b. I/P -> Start the Stopwatch and End the Stopwatch
    c. Logic -> Measure the elapsed time between start and end
    d. O/P -> Print the elapsed time.

Intermediate

1. **Cross Game or Tic-Tac-Toe Game** 

    a. Desc -> Write a Program to play a Cross Game or Tic-Tac-Toe Game. Player 1 is the Computer and the Player 2 is the user. Player 1 take Random Cell that is the Column and Row.
    b. I/P -> Take User Input for the Cell i.e. Col and Row to Mark the ‘X’
    c. Logic -> The User or the Computer can only take the unoccupied cell. The Game is played till either wins or till draw...
    d. O/P -> Print the Col and the Cell after every step.
    e. Hint -> The Hints is provided in the Logic. Use Functions for the Logic...

2. **Take a string from user at runtime. Check that using those characters of string is it possible to make a palindrome string? If yes then print the palindrome string.**

    E.g. abcde O/p: palindrome not possible.

    E.g. abcabc O/p: palindrome possible. abccba

Advance

1. **For a linked list perform the following functions.**

    a. Reverse the link in single link list
    b. Reverse the link using recursion
    c. Reverse the link using stack.

Set 2

Basic

1. **Leap Year**
    a. I/P -> Year, ensure it is a 4 digit number.
    b. Logic -> Determine if it is a Leap Year.
    c. O/P -> Print the year is a Leap Year or not.
2. **An Anagram Detection Example**
    a. Desc -> One string is an anagram of another if the second is simply a rearrangement of the first. For example, 'heart' and 'earth' are anagrams...
    b. I/P -> Take 2 Strings as Input such abcd and dcba and Check for Anagrams
    c. O/P -> The Two Strings are Anagram or not....

Intermediate

1. **Stock Report**
    a. Desc -> Write a program to read in Stock Names, Number of Share, Share Price. Print a Stock Report with total value of each Stock and the total value of Stock.
    b. I/P -> N number of Stocks, for Each Stock Read In the Share Name, Number of Share, and Share Price
    c. Logic -> Calculate the value of each stock and the total value
    d. O/P -> Print the Stock Report.
    e. Hint -> Create Stock and Stock Portfolio Class holding the list of Stocks read from the input file. Have functions in the Class to calculate the value of each stock and the value of total stocks
2. **Write a program to reverse the words of the given string.**

    eg. Hello Vaibhav! Welcome to Bridgelabz

    o/p: olleH !vahbiaV emocleW ot zbalegdirB

Advance

1. **Write a program to perform the following operations on a circular linked list.**
    a. Add a new element at the appropriate position in a linked list.
    b. Search an element in the linked list and delete it.

Set 3

Basic

1. **Flip Coin and print percentage of Heads and Tails**
    a. I/P -> The number of times to Flip Coin. *Ensure it is positive integer*.
    b. Logic -> Use Random Function to get value between 0 and 1. If < 0.5 then tails or heads
    c. O/P -> Percentage of Head vs Tails

2. **Coupon Numbers**
    a. Desc -> Given N distinct Coupon Numbers, how many random numbers do you need to generate distinct coupon number? This program simulates this random process.
    b. I/P -> N Distinct Coupon Number
    c. Logic -> repeatedly choose a random number and check whether it's a new one.
    d. O/P -> total random number needed to have all distinct numbers.
    e. Functions => Write Class Static Functions to generate random number and to process distinct coupons.

Intermediate

1. **Question to find your number**
    a. Desc -> takes a command-line argument N, asks you to think of a number between 0 and N-1, where N = 2^n, and always guesses the answer with n questions.
    b. I/P -> the Number N and then recursively ask true/false if the number is between a high and low value
    c. Logic -> Use Binary Search to find the number
    d. O/P -> Print the intermediary number and the final answer

2. **Create a menu driven program (using switch case statements) to perform the following operations.**
    a. binarySearch method for String
    b. insertionSort method for String
    c. bubbleSort method for String

Advance

1. **Write a program to perform the following operations on a circular linked list.**
    a. Add a new element at the appropriate position in a linked list.
    b. Search an element in the linked list and delete it.

Set 4

Basic

1. **Power of 2**
    a. Desc -> This program takes a command-line argument N and prints a table of the powers of 2 that are less than or equal to 2^N.
    b. I/P -> The Power Value N. *Only works if 0 <= N < 31 since 2^31 overflows an int*
    c. Logic -> repeat until i equals N.
    d. O/P -> Print the year is a Leap Year or not.

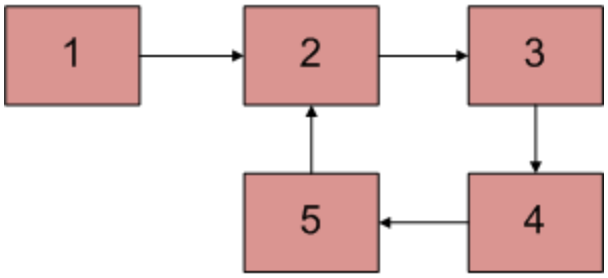
2. **Gambler**
    a. Desc -> Simulates a gambler who start with $stake and place fair $1 bets until he/she goes broke (i.e. has no money) or reach $goal. Keeps track of the number of times he/she wins and the number of bets he/she makes. Run the experiment N times, averages the results, and prints them out.
    b. I/P -> $Stake, $Goal and Number of times
    c. Logic -> Play till the gambler is broke or has won
    d. O/P -> Print Number of Wins and Percentage of Win and Loss.

Intermediate

1. **Binary Search the Word from Word List**
    a. Desc -> Read in a list of words from File. Then prompt the user to enter a word to search the list. The program reports if the search word is found in the list.
    b. I/P -> read in the list words comma separated from a File and then enter the word to be searched
    c. Logic -> Use Arrays to sort the word list and then do the binary search
    d. O/P -> Print the result if the word is found or not

Advance

1. **Detect the loop in given linked list and remove it.**



Set 5

Basic

1. **Harmonic Number**
    a. Desc -> Prints the Nth harmonic number: 1/1 + 1/2 + ... + 1/N (http://users.encs.concordia.ca/~chvatal/notes/harmonic.html).
    b. I/P -> The Harmonic Value N. *Ensure N != 0*
    c. Logic -> compute 1/1 + 1/2 + 1/3 + ... + 1/N
    d. O/P -> Print the Nth Harmonic Value.

2. **Sum of three Integer adds to ZERO**
    a. Desc -> A program with cubic running time. Read in N integers and counts the number of triples that sum to exactly 0.
    b. I/P -> N number of integer, and N integer input array
    c. Logic -> Find distinct triples (i, j, k) such that a[i] + a[j] + a[k] = 0
    d. O/P -> One Output is number of distinct triplets as well as the second output is to print the distinct trniplets.

Intermediate

1. **Insertion Sort**
    a. Desc -> Reads in strings from standard input and prints them in sorted order. Uses insertion sort.
    b. I/P -> read in the list words
    c. Logic -> Use Insertion Sort to sort the words in the String array
    d. O/P -> Print the Sorted List

Advance

1. **Write a program to perform the following singly linked list operations**
    a. insert the node at the end of linked list
    b. insert the node at the beginning of linked list
    c. Insert the node at a particular position in linked list
    d. delete the node at the end of linked list
    e. delete the node at the beginning of linked list
    f. delete the node at a particular position in linked list
    g. Display the elements of the linked list