**NAME: LOGESHWARAN S DATE: 25-03-2023**

**TECHNICAL ASSIGNMENT - 1**

**EXERCISE 1:**

**FIND THE MAXIMUM ELEMENT IN THE ARRAY**

**ALGORITHMS:**

**PRE-CONDITION:**

**Start**

Create a initial set of array elements, that is **arr[ ] = 0**

Initializing **i , temp = 0 ,**

Here **“i”** is a variable that act in a loop for a index value and **temp = 0** will be taken as a temporary value for this condition

**POST-CONDITION:**

In the end of the loop condition,

All the element in the set of array are compare with nearby element and finally resulted the maximum element of the array , using a temp value.

**LOOP-VARIANT:**

In the loop,

When the array value **“arr( i )”** in the given set of the array is than the loop condition less than the temp value **[ arr(i)>temp) is false ]** than the loop is stop

And again next array element will get into the loop

**LOOP-INVARIANT:**

In the loop ,

The array value ( i ) is greater than the previous stores array value “arr( i )”, then it is true.

That final element of the array , is taken as the maximum element .

This is the result of the given array element .

**End**.

**PSEUDOCODE :**

Int i , arr [ ] = {value} , temp=0 ;

For

( i=0 ; i < arr.length ; i++)

{

If ( arr[ i ] > temp )

{

temp = arr [ i ] ;

}

}

**EXERCISE 2:**

**CONVERT ROMAN LETTERS TO INTEGERS**

**ALGORITHMS:**

**PRE-CONDITION:**

**Start.**

Initiate a value for every roman letter which given in the list using char, integer, string.

Here using a **“if – condition”** assign the return value for the every given roman numerals and stores in **“ r ”.**

Initialize **result = 0**.

**POST-CONDITION:**

After end of the loop condition,

Every given characters are assigned each individual values.

In result, all entered roman characters will be display as number values.

**LOOP VARIANT AND INVARIANT:**

Hereget value for **r1 and r2** (r=roman) from the if-condition.

Next, Using loop

Compare r1 and r2 value with **(r1 >= r2)** condition, If this condition is true,

then add the **(r1+r2)** value and get the the result and stores.

If the condition is false, then subtract **(r2-r1)** value and get the result and stores.

When r1 and r2 are condition statisfied for **(r1 >= r2)** , then the result (**res**) will be shown in output.

**End.**

**PSEUDOCODE :**

int value (char r) , n ;

{

If ( r == “ “)

Return value();

Return 0;

}

int romtoint( String str )

{

int sum = 0;

for ( int i = 0; i<str.length(); i++)

{

int r1 = value(str.CharAt(i));

if (i+1 <str.length())

{

int r2 = value(str.CharAt(i));

if (r1 >= r2)

{

       res = res + r1;

     }

else

{

        rest = res - r1;

     }

  }

else

{

         res = res + r1;

      }

   }

Return res;

 }

}

**EXERCISE 3:**

**MOVES ZEROS TO THE END OF THE ARRAY**

**ALGORITHMS:**

**PRE-CONDITION:**

Start

Create the initial set of array elements, that is **arr [ ] = 0**

Initializing **count = 0,**

**POST-CONDITION:**

In the end of the loop condition,

All the element in the **array[ ]** are splits , then all **non-zero** elements are order from **a[0]** and remaining **zeroes** elements are followed by the non-zeroes order at the end of the array.

**LOOP VARIANT AND INVARIANT:**

In the loop conditions,

All the elements in the array are get into the loop and follow the condition **(arr[i] != 0)**

If this condition is **true**, then the non-zero elements are puts into a new array **arr[ i ].** And **count++** till the end of the array.

Or this condition is **false**, in-while condition **(count=0)** the zero elements are counted followed by the elements in the end of the array.

All the **arr()** values a get by **data** , finally the collected data will print in output.

At the result, in the array the **non-zero elements are in right side** and **zero elements are in the left side** of the final array.

**PSEUDOCODE :**

Int arr[] , count = 0

{

for

{

(i = 0 ; i < arr.len ; i++)

{

if (arr[i] != 0)

arr[count++] = arr[i];

}

}

While (count < arr.len)

{

arr[count++] = 0;

}

For ( int data: arr )

}

}

**T H E E N D**