TCS DIGITAL PRACTICE QUESTIONS

Question 1:

Write a program to find the Nth smallest element from a collection.

Input Format

The first input contains integer value X, used to define the size of an array. The second input contains X unsorted integer numbers separated by newline, I.e. A[i].

The third Input contains the value of N, to find the Nth smallest element from the array.

Output Format

The output should be an integer value, the Nth smallest element in the array.

Constraints 1 <= X <= 10 1 <= A[i] <= 1000 1 <= N <= X

Sample1: Input: 5 10 20 40 30 60 3 Output: 30

Explanation: Here, the first input is 5, the size of the array, 10, 20, 40, 30, 60 are the elements of array A[i]. The last input, i.e. 3, is pointing to the 3rd smallest element in the given array. Hence, the 3rd smallest element is 30.

Code:

```
#include <bits/stdc++.h>

using namespace std;

int main()

{

   int n,N=0;

   cin>>n;

   int arr[n];

   for(int i=0;i<n;i++)

   {

      cin>>arr[i];

   }
```

```
    sort(arr, arr+n);
    for(int i=0;i<n;i++)
    {
       cout<<arr[i]<<"  ";
    }
    cout<<"Enter the nth largest element"<<endl;
    cin>>N;
    cout<<arr[N-1]<<endl;


    return 0;
}
```

Question 2:

John and Linda are playing a numbers game. John asked Linda to find the number whose square ends with the number itself. The number should also be a positive integer. Write a program to implement the above logic.

Input Format

Input contains an integer 'N' denoting the number.

Output Format

If the number whose square ends with the number itself, print "Correct Number", otherwise print "Incorrect Number". If the user enters negative integer, the result should display "Wrong Input".

Constraints

1<=N<=10^8

Explanation: Sample Input 1: 5 is a correct number because the square of 5 is 25. The last digit of 25 is 5, which is the same as the given number. Sample Input 2:9 is an incorrect number because the square of 9 is 81. The last digit of 81 is 1, which is not the same as the given number. Sample Input 3:-6 is a negative number. Hence it should print "Wrong Input".

Code:

```
#include <iostream>
#include<string.h>
```

```
#include<cmath>
using namespace std;
int check(int num, int ex)
{
   if(((num*num)%int(pow(10,ex)))==num)
   {
      return 1;
   }
   else
   {
      return 0;
   }
}
int main()
{
   int num,ex,temp;
   cin>>num;
   temp=num;
   ex=to_string(temp).size();
   if(num>=0)
   {
      if(check(num,ex)==1)
         cout<<"Correct number";
      else
         cout<<"Wrong number";
   }
   else
      cout<<"Wrong number";
   return 0;
```

}

Question 3:

Paul is developing an application for a block game. This block game consists of Number blocks,

Uppercase Alphabet blocks, Lowercase Alphabet blocks and some Symbol blocks. Write a program to help Paul identify the precise category of block.

Input Format

First line contains only one character as input of the following category:

Number

Upper Alphabet

Lower Alphabet

Symbol

If the user enters more than one characters, only the first character will be taken into consideration.

Output Format

Output should correctly identify the category of the input entered by the user and print accordingly.

Explanation: Sample Input 1 - User has entered "A" character, which is a capital alphabet. Hence output should print "UPPER ALPHABET". Sample Input 2-User has entered "d" character, which is a lowercase alphabet. Hence output should print "lower alphabet". Sample Input 3-User has entered 9 number. Hence output should print "Number". Sample Input 4 - User has entered "%", which is a symbol. Hence output should print "Symbol". Sample Input 5 and 6-If the user enters any combination of the 4 categories mentioned above, only the first character is to be taken into consideration for validation. In case 5, the first character is in lowercase, hence it should print "lower alphabet" and likewise.

Code:

```
#include <iostream>

#include<string.h>

#include<cmath>

using namespace std;

void check(int ch)

{

   if(isalpha(ch)!=0)
```

```
    {
        if(isupper(ch)!=0)
        {
            cout<<"Upper";
        }
        else
        {
            cout<<"Lower";
        }
    }
    else if(isdigit(ch)!=0)
    {
        cout<<"Number";
    }
    else
    {
        cout<<"Symbol";
    }
}
int main()
{
    string str;
    cin>>str;
    check(str[0]);
    return 0;
}
```