ISHITA SINGH

1)A)

#include<bits/stdc++.h>

using namespace std;

int fib(int n)

{

if (n <= 1)

return n;

return fib(n-1) + fib(n-2);

}

int main ()

{

int n ;

cin>>n;

cout << fib(n);

getchar();

return 0;

}

1. B)

#include <iostream>

using namespace std;

int factorial( int n)

{

if (n == 0)

return 1;

return n \* factorial(n - 1);

}

int main()

{

int num ;

cin>>num;

cout << "Factorial of "

<< num << " is " << factorial(num) << endl;

return 0;

}

1. C)

#include <iostream>

using namespace std;

int search(int arr[], int n, int x)

{

int i;

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

if (arr[i] == x)

return i;

return -1;

}

int main(void)

{

int n,x;

cin>>n;

cin>>x;

int arr[n];

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

cin>>arr[i];

int result = search(arr, n, x);

(result == -1)? cout<<"Element is not present in array"

: cout<<"Element is present at index " <<result;

return 0;

}

2)

#include<iostream>

using namespace std;

int main()

{

int i,n,large,small;

cout << "Enter the number of elements :";

cin >> n;

int a[n];

cout << "Input the array elements : ";

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

cin >> a[i];

large=small=a[0];

for(i=1;i<n;++i)

{

if(a[i]>large)

large=a[i];

if(a[i]<small)

small=a[i];

}

cout << "The smallest element is " << small << endl;

cout << "The largest element is "<< large << endl;

return 0;

}

3)

#include<iostream>

#include<climits>

using namespace std;

int maxSubArraySum(int a[], int size)

{

int max = INT\_MIN, max1 = 0;

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

{

max1 = max1 + a[i];

if (max < max1)

max = max1;

if (max1 < 0)

max1 = 0;

}

return max;

}

int main()

{

int n;

cin>>n;

int a[n];

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

cin>>a[i];

int maxsum = maxSubArraySum(a, n);

cout << "Maximum contiguous sum is " << maxsum;

return 0;

}

4)

#include <bits/stdc++.h>

using namespace std;

int Pairs(int arr[], int n, int sum)

{

int count = 0;

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

for (int j = i + 1; j < n; j++)

if (arr[i] + arr[j] == sum)

cout << arr[i]<<"and"<<arr[j]<<"are pairs" << endl;

}

int main()

{

int n,sum;

cin>>n;

cin>>sum;

int arr[n];

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

cin>>arr[i];

Pairs(arr, n, sum);

return 0;

}

5)

#include <iostream>

#include <string.h>

using namespace std;

int main()

{

char str1[20], str2[20];

int i, j, len = 0, flag = 0;

cout << "Enter the string : ";

gets(str1);

len = strlen(str1) - 1;

for (i = len, j = 0; i >= 0 ; i--, j++)

str2[j] = str1[i];

if (strcmp(str1, str2))

flag = 1;

if (flag == 1)

cout << str1 << " is not a palindrome";

else

cout << str1 << " is a palindrome";

return 0;

}

6) #include<bits/stdc++.h>

#define ASCII\_SIZE 256

using namespace std;

char MaxOccur(char\* str)

{

int count[ASCII\_SIZE] = {0};

int len = strlen(str);

int max = 0;

char result;

for (int i = 0; i < len; i++) {

count[str[i]]++;

if (max < count[str[i]]) {

max = count[str[i]];

result = str[i];

}

}

return result;

}

int main()

{

char str[50];

gets(str);

cout << "Max occurring character is "

<< MaxOccur(str);

}

7)

#include <iostream>

using namespace std;

compressed\_str(string str)

{

int len = str.length();

for (int i = 0; i < len; i++) {

int count = 1;

while (i < len - 1 && str[i] == str[i + 1]) {

count++;

i++;

}

if (count == 1)

{

cout << str[i];

}

else

{

cout << str[i] << count;

}

}

}

int main()

{

char str[50];

gets(str);

compressed\_str(str);

return 0;

}

8) #include <iostream>

using namespace std;

int main()

{

int arr[3][3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

for (int i = 0; i < 3; i++)

for (int j = 0; j < 3; j++)

{

cout<<arr[j][i]<<" ";

}

return 0;

}

9)

#include<iostream>

using namespace std;

int main()

{

int m,n;

cout<<"Enter the no. of rows and columns"<<endl;

cin>>m>>n;

int a[m][n];

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

{

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

{

cin>>a[i][j];

}

}

cout<<"Enter the no. to be searched"<<endl;

int x;

cin>>x;

int i = 0;

int j = n-1;

int flag= 0;

while(i<m&&j<n)

{

if(a[i][j]==x)

{

flag = 1;

break;

}

else if(x<a[i][j])

{ j--;

}

else

{

i++;

}

}

if (flag==1)

cout<<"No. found at row "<<i+1<<"column "<<j+1<<endl;

else

cout<<"not found"<<endl;

return 0;

}