#### ADA Lab-1

1 a) Sort a given set of elements using Selection sort.

###### Code:

#include<iostream>

using namespace std;

int count = 0;

void selection\_sort(int \*a ,int n){

    for(int i=0; i<n-1; i++)

    { int m =i;

        for (int j=i+1; j<n; j++){

         count++;

         if(a[j]<a[m])

          m = j;

        }

        swap(a[i],a[m]);

    }

}

int main()

{

    int n;

    cout<<"Enter size of array\n";

    cin>>n;

    int arr[n];

    cout<<"Enter "<<n<<" numbers\n";

    for(int i=0;i<n;i++){

        cin>>arr[i];

    }

    cout<<"Entered array is : ";

    for(int i=0;i<n;i++){

        cout<< arr[i]<<" ";

    }

    cout<<endl;

    selection\_sort(arr,n);

    cout<<"Sorted array is : ";

    for(int i=0;i<n;i++){

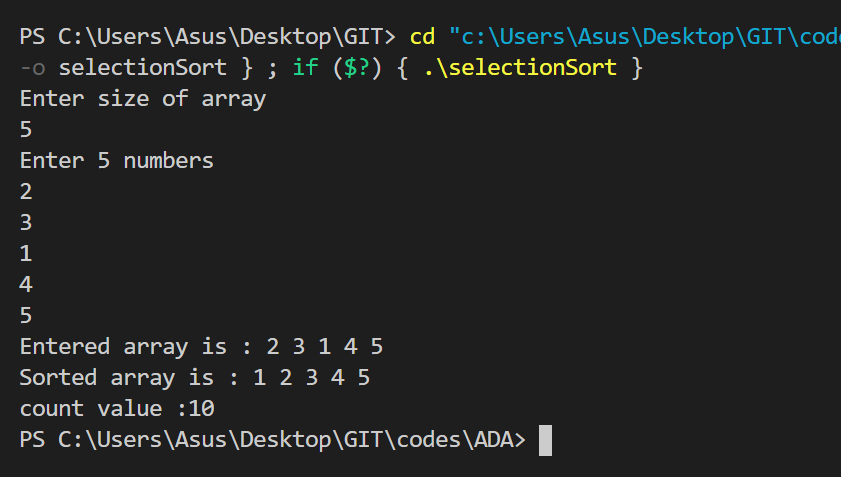
        cout<< arr[i]<<" ";

    }

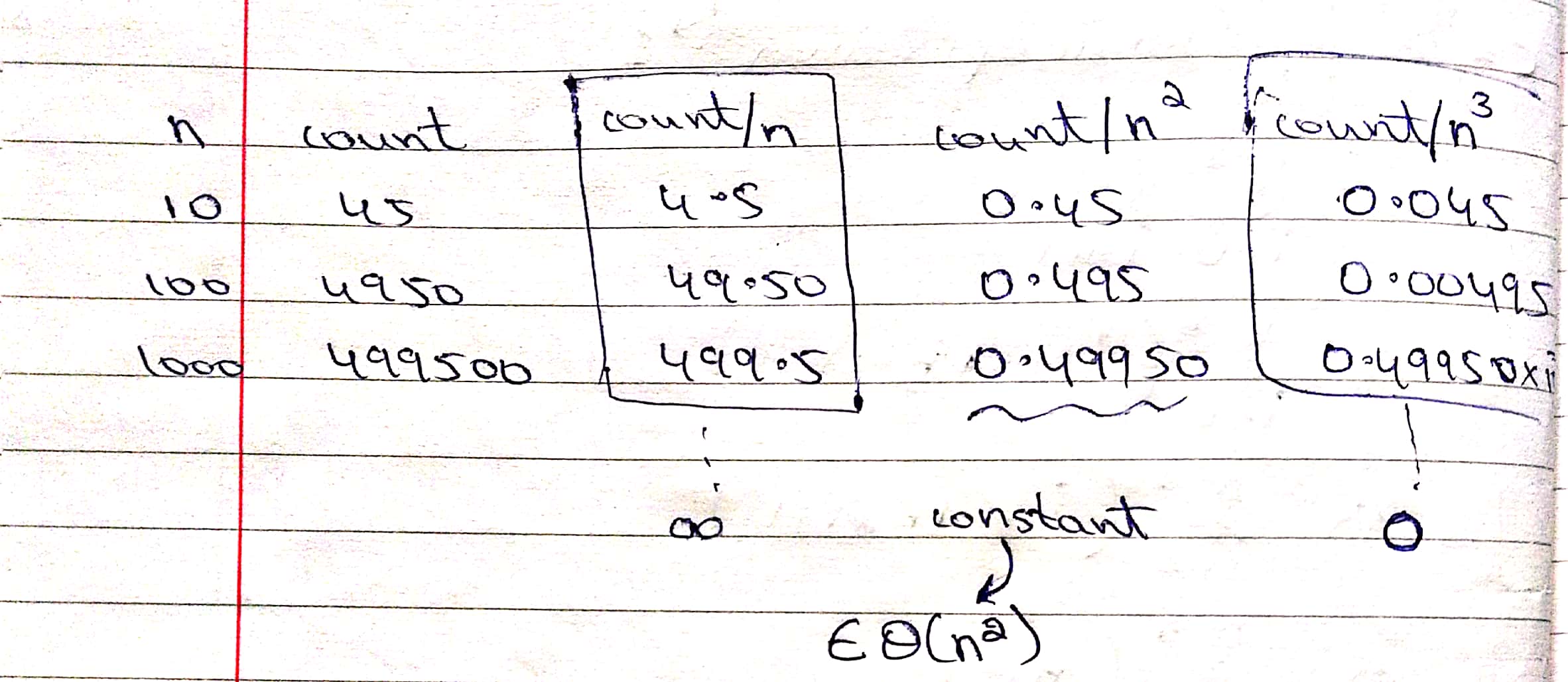
    cout<<"\ncount value :"<<count;

}

###### Output:



###### Analysis:



1 b) Implement Brute force String matching algorithm.

###### Code:

#include<iostream>

#include<string.h>

int count=0;

using namespace std;

void strMatching(string t ,string p)

{

  int x = t.length();

  int y = p.length();

int flag=0;

  for(int i = 0;i<=x-y;i++)

  {   int j;

      for(j=0;j<y;j++)

      {   count++;

          if(t[i+j] == p[j])

           continue;

        else

           break;

      }

      if(j==y)

       {

           flag++;

           cout<<"pattern found at index "<<i<<endl;

       }

  }

  if(flag==0){

      cout<<"pattern not found\n";

  }

// cout<<"count/x :"<<count/x<<endl<<"count/(x\*x) : "<<count/(x\*x)<<endl<<"count/y : "<<count/y<<endl<<"count/xy : "<<count/(x\*y)<<endl;

}

int main()

{

    string x,y;

    cout<<"Enter text \n";

    getline(cin,x);

    cout<<"Enter pattern \n";

    getline(cin,y);

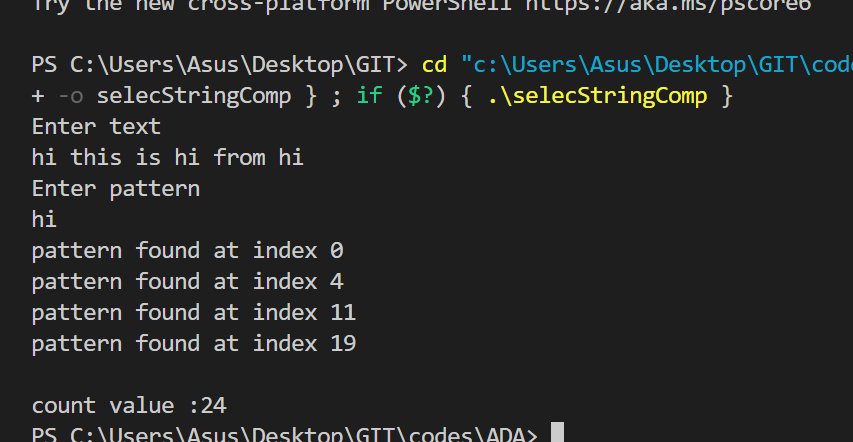
    strMatching(x,y);

    cout<<"\ncount value :"<<count;

    return 0;

}

###### Output:



###### Analysis:

