ESERCIZIO 1:

Complessità O(nlogn)

#include <bits/stdc++.h>

using namespace std;

int occ(int a[], int i, int j, int m)

{ int k=i+(j-i)/2;

int tmp=0;

if ((i > j) || (i == j && a[i] != m)) return 0;

if ( a[i] == m) return 1;

return occ(a, i,k, m) + occ(a, 1+ k,j, m);

}

int main(){

int k;

cin>>k;

while(k--) {

int x;

cin>>x;

int n;

int a[n];

cin>>n;

for(int i=0;i<n;i++) cin>>a[i];

cout<<occ(a, 0, n+1, x)<<'\n';

}

return 0;

}

ESERCIZIO 2

#include <bits/stdc++.h>

using namespace std;

vector<int> v;

vector<int> primi;

bool isPrime(int x)

{ int rad = sqrt(x);

if (x == 1) return false;

for (int i = 2; i <= rad; i++)

if (x % i == 0)

return false;

return true;

}

void stampa()

{

int l = v.size();

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

cout << v[i] << " ";

cout << "\n";

}

void somma(int sum, int N, int S, int i)

{ if (sum == S && v.size() == N)

{

stampa();

return;

}

if (sum > S || i == primi.size())

return;

v.push\_back(primi[i]);

somma(sum+primi[i], N, S, i+1);

v.pop\_back();

somma(sum, N, S, i+1);

}

void ris(int N, int S, int P)

{

for (int i = P+1; i <=S ; i++)

{

if (isPrime(i)) primi.push\_back(i);

}

if (primi.size() < N)

return;

somma(0, N, S, 0);

}

int main(){

int S,N,P,tc=1;

while(cin>>S>>N>>P){

cout<<"CASO DI TEST "<<tc<<'\n';

ris(N, S, P);

tc++;}

return 0;

}