Q2.

Problem Statement:

Given an unsorted array a of size N of non-negative integers,

find a continuous sub-array which adds to a given number sum.

Input Format:

The first line contains an integer, denoting the size of the array.

The second line contains integers denoting the elements

of the array.

The last line contains an integer, denoting the sum.

Constraints

1<= n<=100

1<=arr<= 1000, where arr is the ith element of the array.

1<= n<=100000

Output Format:

The output line contains integers denoting the indexes.

TESTCASE 1:

Input:

7

[1, 4, 0, 0, 3, 10, 5]

sum = 7

Output:

Sum found between indexes 1 and 4

TESTCASE 2:

Input:

2

[1, 4]

sum = 0

Output:

No subarray found

#include <stdio.h>

int main() {

int sumToFind,arr[100],i,j,sum,FlagToindicateNumberFound=0,SizeOfArray;

scanf("%d",&SizeOfArray);

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

{

scanf("%d",arr+i);

}

scanf("%d",&sumToFind);

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

{ sum=0;

for(j=i;j<SizeOfArray;j++)

{

sum+=arr[j];

if(sum==sumToFind)

{

printf("Sum Found Between Indexex %d & %d",i, j);

FlagToindicateNumberFound=1;

break;

}

}

if(FlagToindicateNumberFound==1)

{

break;

}

}

if(FlagToindicateNumberFound==0)

{

printf("No subarray found");

}

return 0;

}