Question-1:

Write a program to return the difference between the count of odd numbers and even numbers.

Note : You are expected to write code in the countOddEvenDifference function only which will receive the first parameter as the number of items in the array and second parameter as the array itself. you are not required to take input from the console.

Example
Finding the difference between the count of odd and even numbers from a list of 5 number

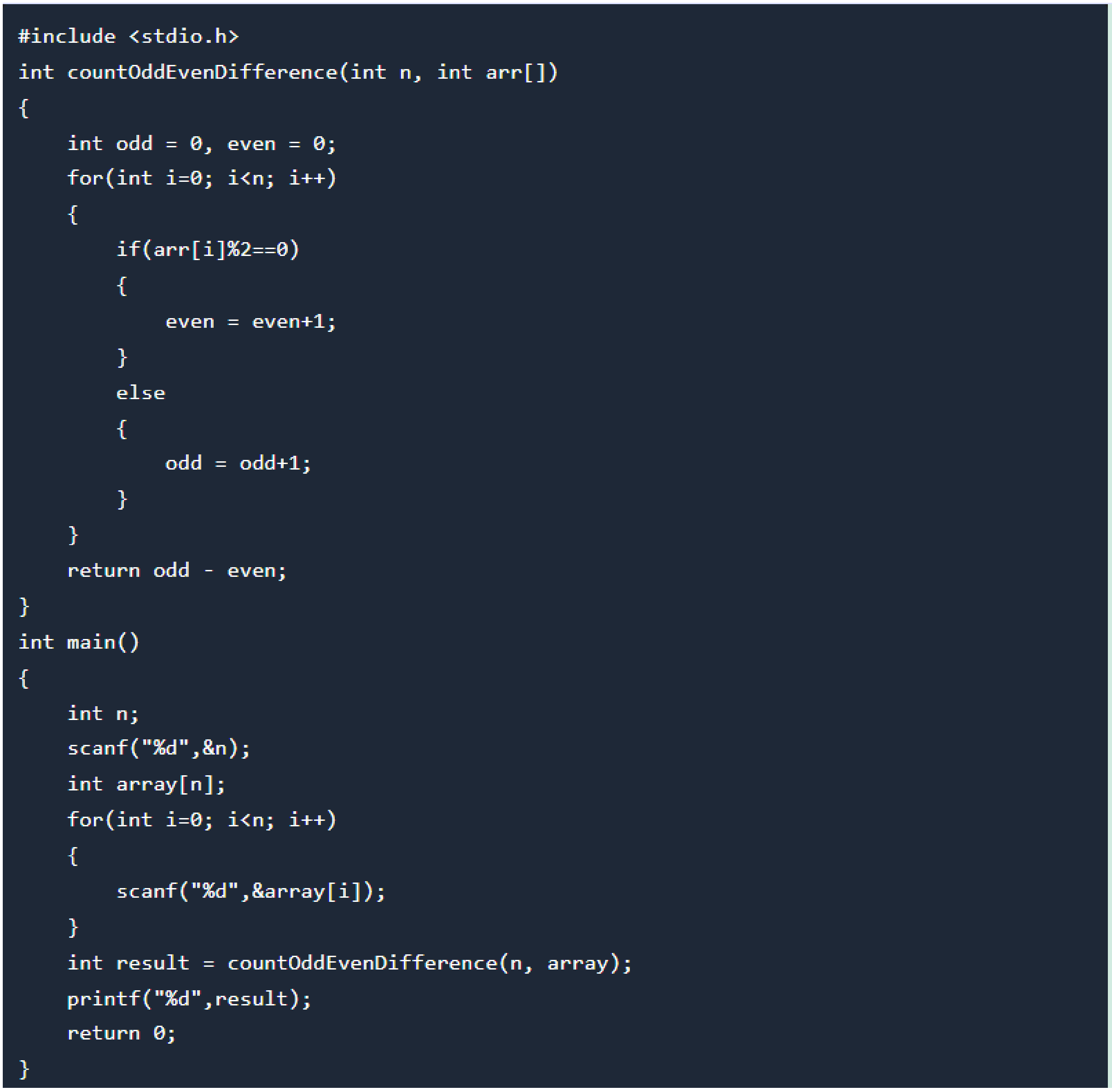
Input
input 1 : 8
input 2 : 10 20 30 40 55 66 77 83

Output
-2

Explanation
The first paramter (8) is the szie of the array. Next is an array of integers. The calculation of difference between count sum of odd and even numbers is as follows:

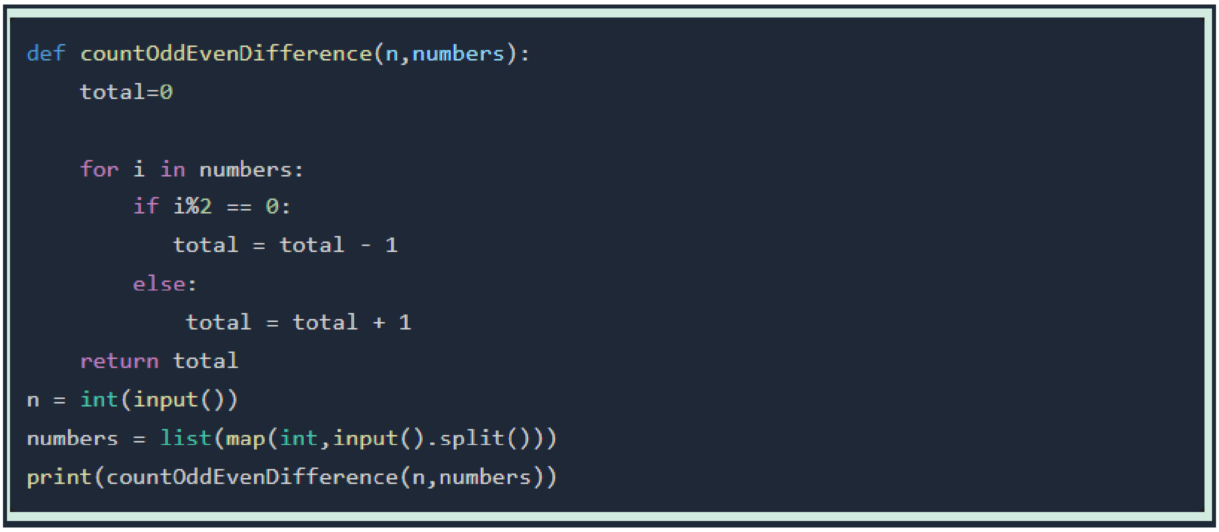
3 (count of odd numbers) – 5 (count of even numbers) = -2

C Code:



```c
#include <stdio.h>
int countOddEvenDifference(int n, int arr[])
{
    int odd = 0, even = 0;
    for(int i=0; i<n; i++)
    {
        if(arr[i]%2==0)
        {
            even = even+1;
        }
        else
        {
            odd = odd+1;
        }
    }
    return odd - even;
}
int main()
{
    int n;
    scanf("%d",&n);
    int array[n];
    for(int i=0; i<n; i++)
    {
        scanf("%d",&array[i]);
    }
    int result = countOddEvenDifference(n, array);
    printf("%d",result);
    return 0;
}
```

Python code:



```
def countOddEvenDifference(n,numbers):
    total=0

    for i in numbers:
        if i%2 == 0:
           total = total - 1
        else:
            total = total + 1
    return total
n = int(input())
numbers = list(map(int,input().split()))
print(countOddEvenDifference(n,numbers))
```

Question-2:

Write a program to calculate and return the sum of absolute difference between the adjacent number in an array of positive integers from the position entered by the user.

Note : You are expected to write code in the findTotalSum function only which receive three positional arguments:

1st : number of elements in the array
2nd : array
3rd : position from where the sum is to be calculated

Example

Input
input 1 : 7
input 2 : 11 22 12 24 13 26 14
input 3 : 5

Output
25

Explanation

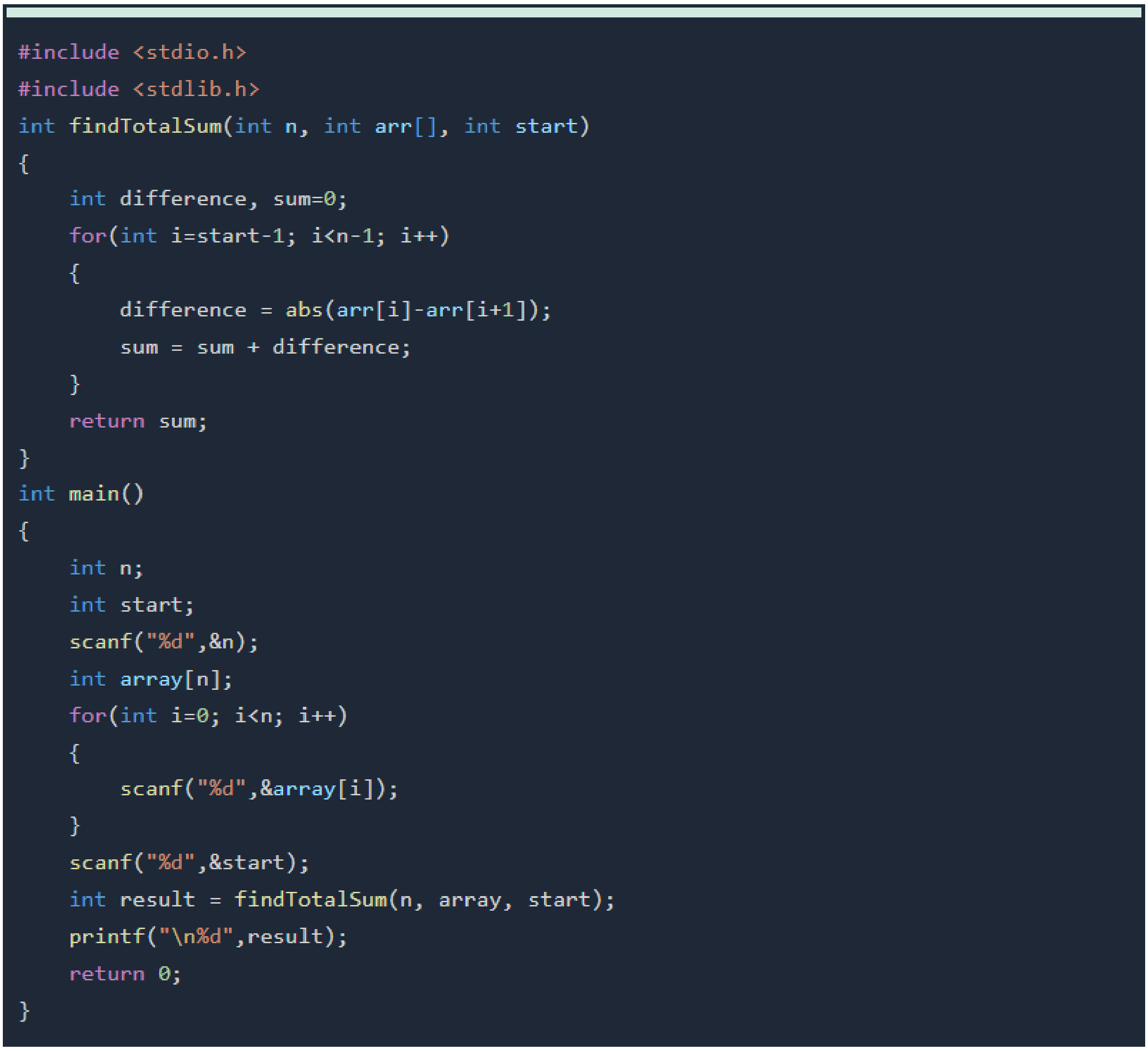
The first parameter 7 is the size of the array. Next is an array of integers and input 5 is the position from where you have to calculate the Total Sum. The output is 25 as per calculation below.

| 26-13 | = 13

| 14-26 | = 12

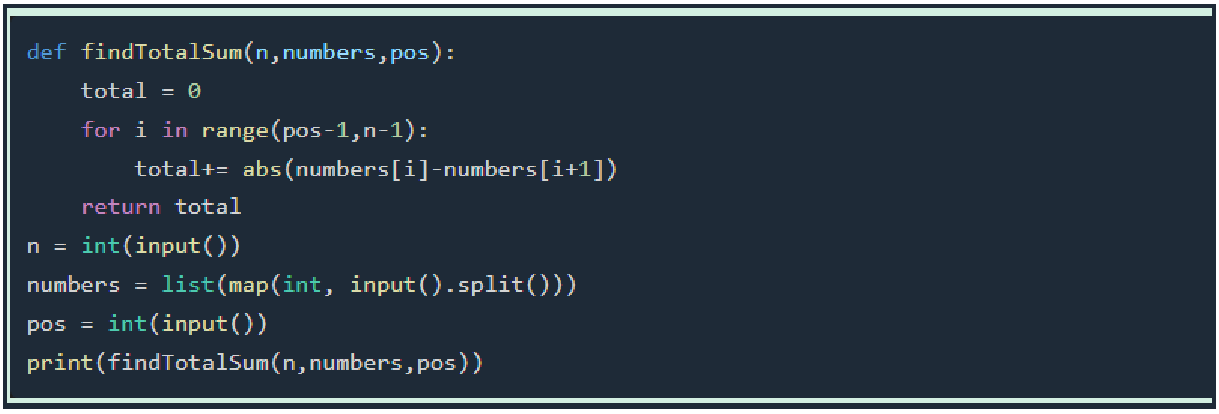
Total Sum = 13 + 12 = 25

C Code:



```c
#include <stdio.h>
#include <stdlib.h>
int findTotalSum(int n, int arr[], int start)
{
    int difference, sum=0;
    for(int i=start-1; i<n-1; i++)
    {
        difference = abs(arr[i]-arr[i+1]);
        sum = sum + difference;
    }
    return sum;
}
int main()
{
    int n;
    int start;
    scanf("%d",&n);
    int array[n];
    for(int i=0; i<n; i++)
    {
        scanf("%d",&array[i]);
    }
    scanf("%d",&start);
    int result = findTotalSum(n, array, start);
    printf("\n%d",result);
    return 0;
}
```

Python code:



```
def findTotalSum(n,numbers,pos):
    total = 0
    for i in range(pos-1,n-1):
        total+= abs(numbers[i]-numbers[i+1])
    return total
n = int(input())
numbers = list(map(int, input().split()))
pos = int(input())
print(findTotalSum(n,numbers,pos))
```

Question-3:

Write a program to find the difference between the elements at odd index and even index.

Note : You are expected to write code in the findDifference function only which receive the first parameter as the numbers of items in the array and second parameter as the array itself. You are not required to take the input from the console.

Example

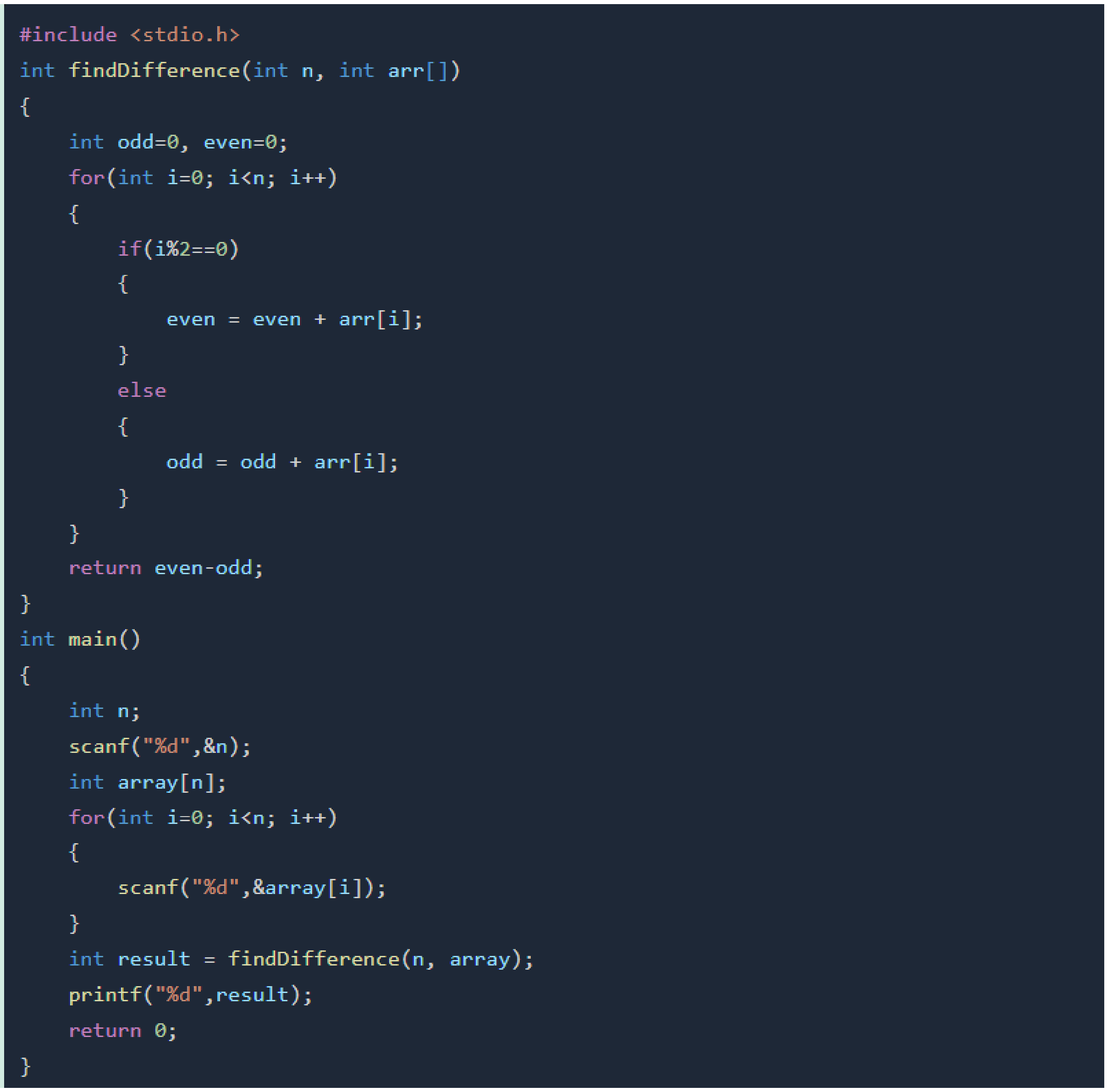
Finding the maximum difference between adjacent items of a list of 5 numbers

Input
input 1 : 7
input 2 : 10 20 30 40 50 60 70

Output
40

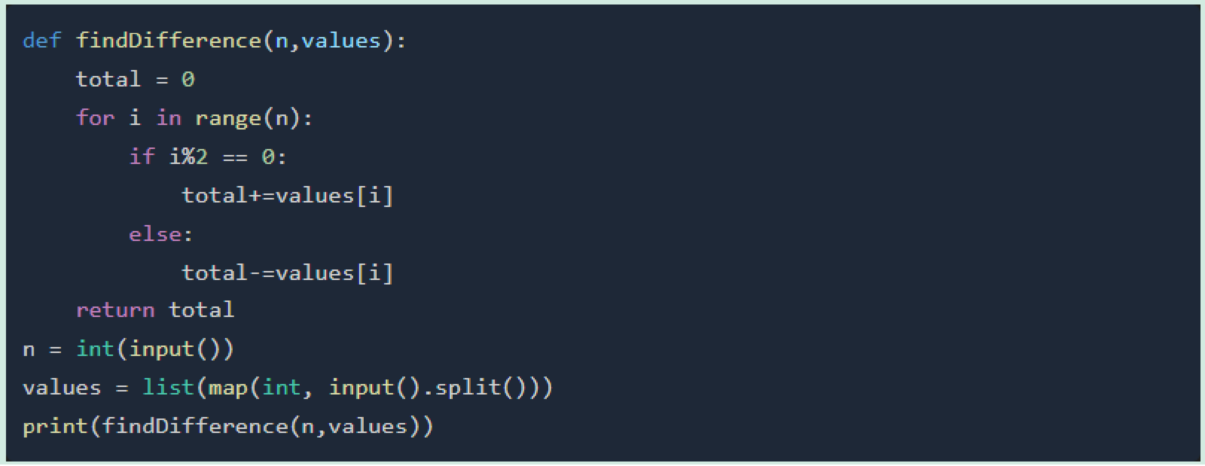
Explanation
The first parameter 7 is the size of the array. Sum of element at even index of array is 10 + 30 + 50 + 70 = 160 and sum of elements at odd index of array is 20 + 40 + 60 = 120. The difference between both is 40

C Code:



```c
#include <stdio.h>
int findDifference(int n, int arr[])
{
    int odd=0, even=0;
    for(int i=0; i<n; i++)
    {
        if(i%2==0)
        {
            even = even + arr[i];
        }
        else
        {
            odd = odd + arr[i];
        }
    }
    return even-odd;
}
int main()
{
    int n;
    scanf("%d",&n);
    int array[n];
    for(int i=0; i<n; i++)
    {
        scanf("%d",&array[i]);
    }
    int result = findDifference(n, array);
    printf("%d",result);
    return 0;
}
```

Python Code:



```python
def findDifference(n,values):
    total = 0
    for i in range(n):
        if i%2 == 0:
            total+=values[i]
        else:
            total-=values[i]
    return total
n = int(input())
values = list(map(int, input().split()))
print(findDifference(n,values))
```

Question-4:

A Cloth merchant has some pieces of cloth of different lengths. He has an order of curtains of length of 12 feet. He has to find how many curtains can be made from these pieces. Length of pieces of cloth is recorded in feet.

Note : You are expected to write code in the findTotalCurtains function only which receive the first parameter as the number of items in the array and second parameter as the array itself. You are not required to take the input from the console.

Example

Finding the total curtains from a list of 5 cloth pieces.

Input

input 1 : 5

input 2 : 3 42 60 6 14

Output

9

Explanation

The first parameter 5 is the size of the array. Next is an array of measurements in feet. The total number of curtains is 5 which is calculated as under

3 -> 0

42 -> 3

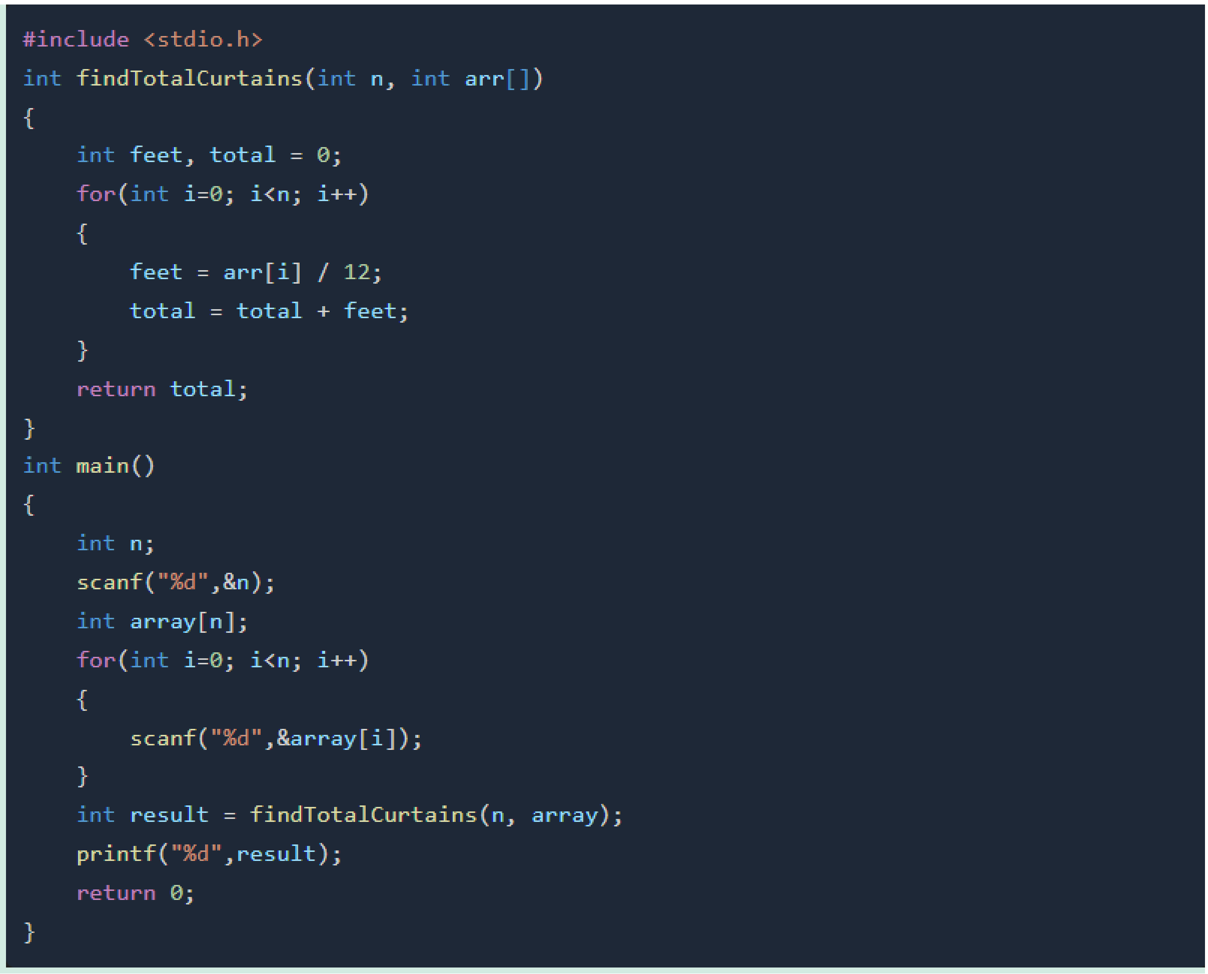
60 -> 5

6 -> 0

14 -> 1

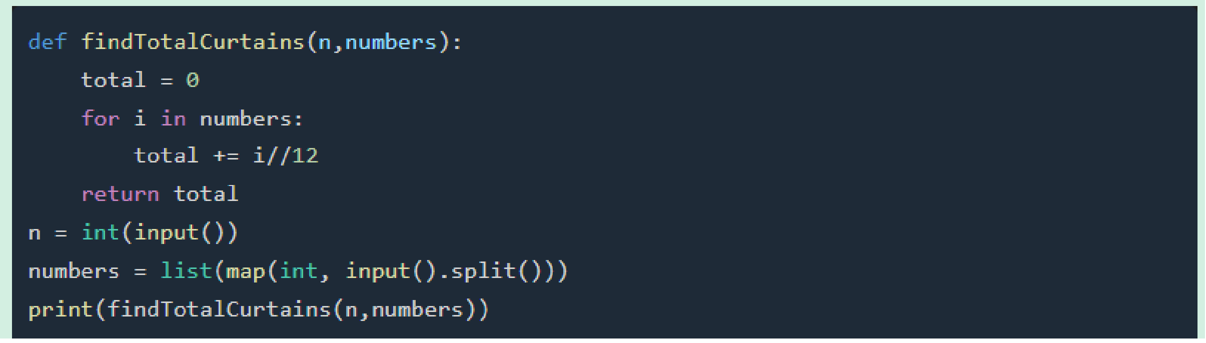
total = 9

C Code:



```c
#include <stdio.h>
int findTotalCurtains(int n, int arr[])
{
    int feet, total = 0;
    for(int i=0; i<n; i++)
    {
        feet = arr[i] / 12;
        total = total + feet;
    }
    return total;
}
int main()
{
    int n;
    scanf("%d",&n);
    int array[n];
    for(int i=0; i<n; i++)
    {
        scanf("%d",&array[i]);
    }
    int result = findTotalCurtains(n, array);
    printf("%d",result);
    return 0;
}
```

Python Code:



```python
def findTotalCurtains(n,numbers):
    total = 0
    for i in numbers:
        total += i//12
    return total
n = int(input())
numbers = list(map(int, input().split()))
print(findTotalCurtains(n,numbers))
```

Question-5:

k th largest factor of N

Problem Description:

A positive integer d is said to be a factor of another positive integer N if when N is divided by d, the remainder obtained is zero. For example, for number 12, there are 6 factors 1, 2, 3, 4, 6, 12. Every positive integer k has at least two factors, 1 and the number k itself. Given two positive integers N and k, write a program to print the kth largest factor of N.

Input Format: The input is a comma-separated list of positive integer pairs (N, k).

Output Format: The kth highest factor of N. If N does not have k factors, the output should be 1.

Constraints:

- 1<N<10000000000
- 1<k<600.

You can assume that N will have no prime factors which are larger than 13.

Example 1

- Input: 12,3
- Output: 4

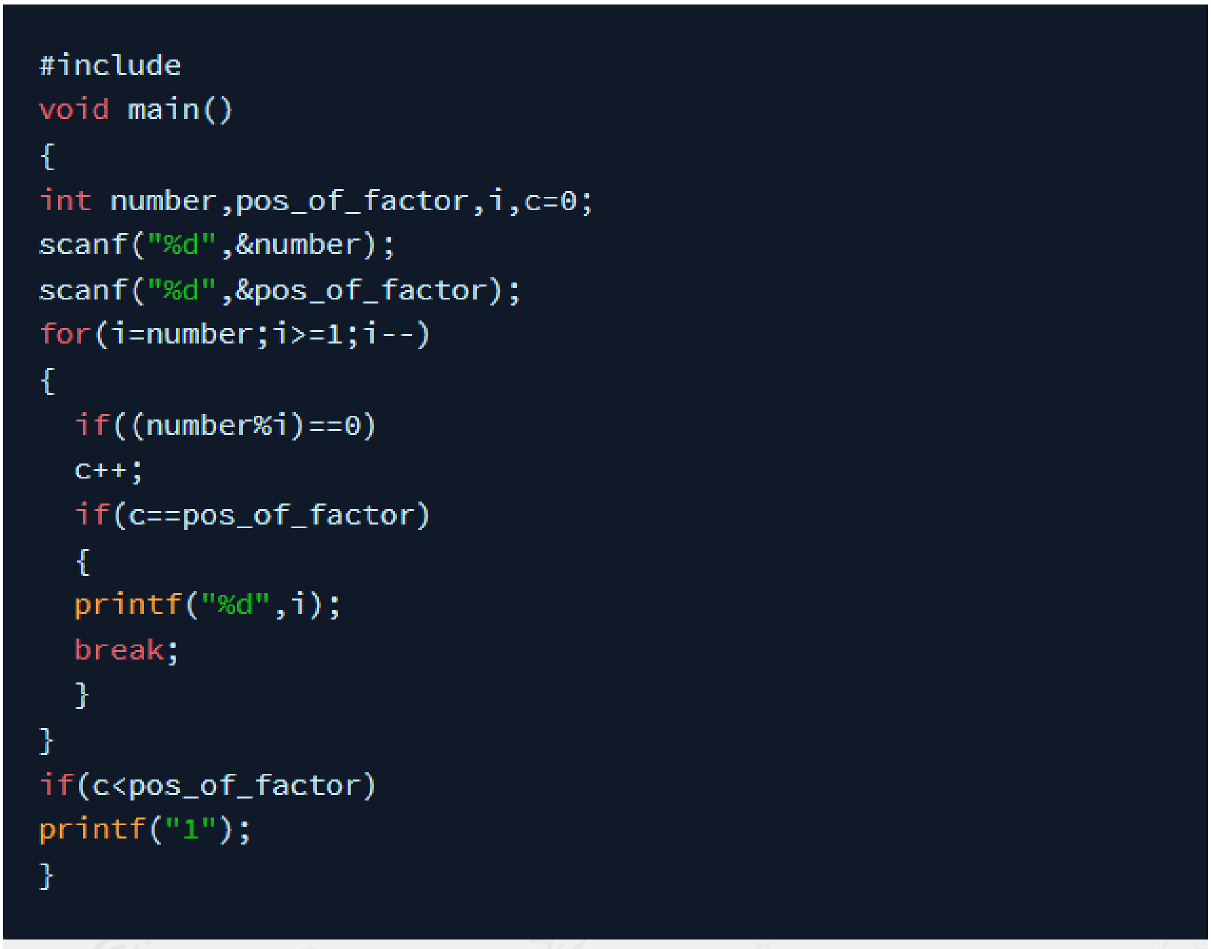
Explanation: N is 12, k is 3. The factors of 12 are (1,2,3,4,6,12). The highest factor is 12 and the third largest factor is 4. The output must be 4.

Example 2

- Input: 30,9
- Output: 1

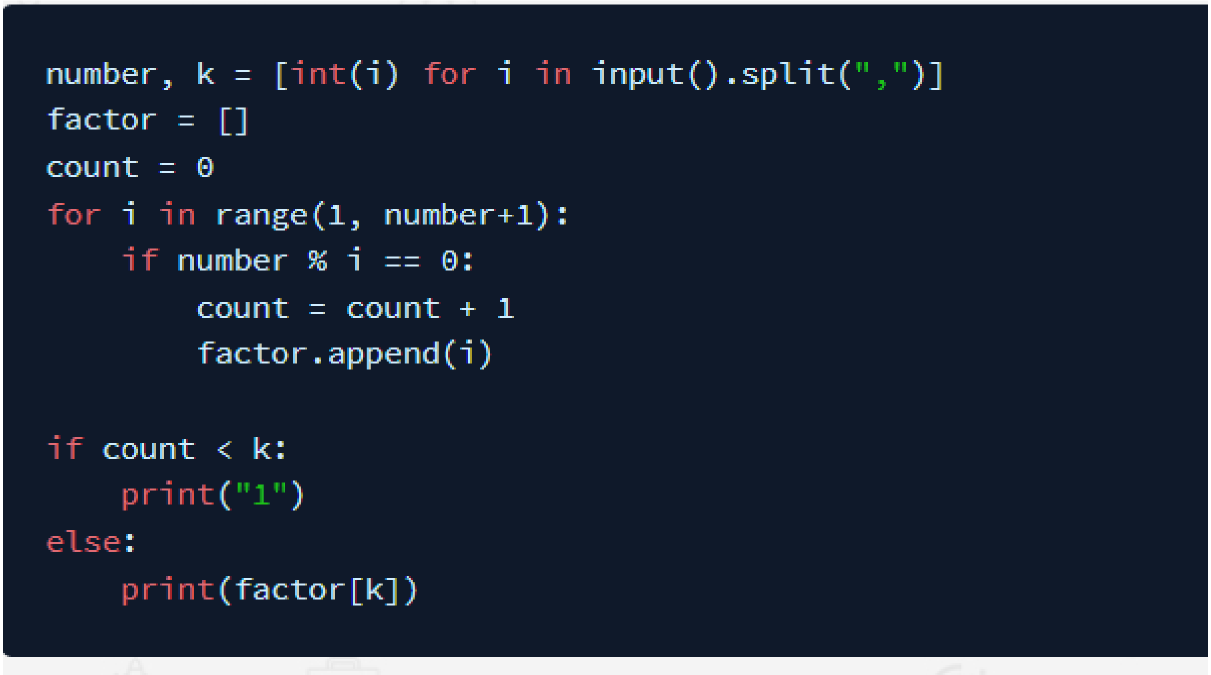
Explanation: N is 30, k is 9. The factors of 30 are (1,2,3,5,6,10,15,30). There are only 8 factors. As k is more than the number of factors, the output is 1.

C Code:



```
#include
void main()
{
int number,pos_of_factor,i,c=0;
scanf("%d",&number);
scanf("%d",&pos_of_factor);
for(i=number;i>=1;i--)
{
  if((number%i)==0)
  c++;
  if(c==pos_of_factor)
  {
  printf("%d",i);
  break;
  }
}
if(c<pos_of_factor)
printf("1");
}
```

Python Code:



```python
number, k = [int(i) for i in input().split(",")]
factor = []
count = 0
for i in range(1, number+1):
    if number % i == 0:
        count = count + 1
        factor.append(i)

if count < k:
    print("1")
else:
    print(factor[k])
```