

سلسلة الخوارزميات وحل المشاكل - المستوى الثاني



26+ Years  
of Experience

# PROGRAMMING ADVICES

LEARN THE  
RIGHT WAY

**Mohammed Abu-Hadhoud**

MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITIL®, MCPD, MCSD



حقوق النشر محفوظة، أسعار الكورسات في المنصة هي أسعار  
رمزية جدا، ارجو عدم نشر هذه الوثيقة لان نشرها سيمنعنا من  
الاستمرار في تقديم العلم للآخرين

ارجو عدم استخدام هذه الوثيقة من غير وجه حق لأنك ستحرم الاف  
الناس من التعلم

**ProgrammingAdVICES.com**



## Problem # 29/2 Solution Using C++

```
#include <iostream>
using namespace std;

enum enPrimNotPrime { Prime = 1, NotPrime = 2 };

enPrimNotPrime CheckPrime(int Number)
{
    int M = round(Number / 2);

    for (int Counter = 2; Counter <= M; Counter++)
    {
        if (Number % Counter == 0)
            return enPrimNotPrime::NotPrime;
    }

    return enPrimNotPrime::Prime;
}

int RandomNumber(int From, int To)
{
    //Function to generate a random number
    int randNum = rand() % (To - From + 1) + From;
    return randNum;
}

void FillArrayWithRandomNumbers(int arr[100], int& arrLength)
{
    cout << "\nEnter number of elements:\n";
    cin >> arrLength;

    for (int i = 0; i < arrLength; i++)
        arr[i] = RandomNumber(1, 100);
}
```



## Problem # 29/2 Solution Using C++

```
void CopyOnlyPrimeNumbers(int arrSource[100], int
arrDestination[100], int arrLength, int& arr2Lenght)
{
    int Counter = 0;
    for (int i = 0; i < arrLength; i++)
    {
        if (CheckPrime(arrSource[i]) == enPrimNotPrime::Prime)
        {
            arrDestination[Counter] = arrSource[i];
            Counter++;
        }
    }

    arr2Lenght = --Counter;
}

void PrintArray(int arr[100], int arrLength)
{
    for (int i = 0; i < arrLength; i++)
        cout << arr[i] << " ";

    cout << "\n";
}

int main() {
    //Seeds the random number generator in C++, called only once
    srand((unsigned)time(NULL));

    int arr[100], arrLength;
    FillArrayWithRandomNumbers(arr, arrLength);

    int arr2[100], arr2Length = 0;
    CopyOnlyPrimeNumbers(arr, arr2, arrLength, arr2Length);

    cout << "\nArray 1 elements:\n";
    PrintArray(arr, arrLength);

    cout << "\nPrime Numbers in Array2:\n";
    PrintArray(arr2, arr2Length);

    return 0;
}
```