

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 # 39/2 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);
}

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

    cout << "\n";
}
```



Problem # 39/2 Solution Using C++

```
void AddArrayElement(int Number, int arr[100], int& arrLength)
{
    //its a new element so we need to add the length by 1
    arrLength++;
    arr[arrLength - 1] = Number;
}

void CopyPrimeNumbers(int arrSource[100], int arrDestination[100],
int arrLength, int& arrDestinationLength)
{
    for (int i = 0; i < arrLength; i++)
        if (CheckPrime(arrSource[i]) == enPrimNotPrime::Prime)
        {
            AddArrayElement(arrSource[i], arrDestination,
arrDestinationLength);
        }
}

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

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

    FillArrayWithRandomNumbers(arr, arrLength);

    CopyPrimeNumbers(arr, arr2, arrLength, arr2Length);

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

    cout << "\nArray 2 Prime numbers:\n";
    PrintArray(arr2, arr2Length);

    return 0;
}
```