Practice Arena

Practice problems aimed to improve your coding skills.

- PRACTICE-02_SCAN-PRINT
- PRACTICE-03_TYPES
- LAB-PRAC-02_SCAN-PRINT
- LAB-PRAC-01
- PRACTICE-04_COND
- BONUS-PRAC-02
- LAB-PRAC-03_TYPES
- PRACTICE-05_COND-LOOPS
- LAB-PRAC-04_COND
- LAB-PRAC-05_CONDLOOPS
- PRACTICE-07_LOOPS-ARR
- LAB-PRAC-06_LOOPS
- LAB-PRAC-07_LOOPS-ARR
- LABEXAM-PRAC-01_MIDSEM
- PRACTICE-09_PTR-MAT
- LAB-PRAC-08_ARR-STR
- PRACTICE-10_MAT-FUN
- LAB-PRAC-09_PTR-MAT
- LAB-PRAC-10_MAT-FUN
- PRACTICE-11_FUN-PTR
  - Circular Queue
  - Primes are here again
  - The Clones of the Clones
- LAB-PRAC-11_FUN-PTR
- LAB-PRAC-12_FUN-STRUC
- LABEXAM-PRAC-02_ENDSEM
- LAB-PRAC-13_STRUC-NUM
- LAB-PRAC-14_SORT-MISC

Primes are here again

PRACTICE-11_FUN-PTR

WARNING: This problem is otherwise very simple. The point of this question is to practice writing functions to perform simple operations to write nice code. Use the template provided and practice writing functions.

You will be given a positive number n and then n positive integers. Store these numbers in an array arr. As your output, you have to print n numbers where ith number is arr[i-1]*(sum of all primes strictly less than arr[i-1]) where i runs from 1 to n. For this problem, 0 and 1 are not considered primes.

WARNING: use long variables to perform computations even though the input will only be integer variables. The template is given below as well in case you erase it.

Input Format:
n (integer denoting the size of the array)
n space separated positive integer

Output Format:
n space separated integers

Example:
Input:
1
2

Output:
0

```
#include <stdio.h>

int check_prime(int n){
    if(n < 2) return 0;
    for(int i = 2; i < n; i++)
        if(n % i == 0)
            return 0;
    return 1;
}

long prime_sum(int n){
    long sum = 0;
    for(int i = 2; i < n; i++)
        if(check_prime(i))
            sum += i;
    return sum;
}

int main(){
    int n, i;
    scanf("%d", &n);
    int arr[n];

    for(i = 0; i < n; i++){
        scanf("%d", arr + i);
        printf("%ld", arr[i]*prime_sum(arr[i]));
        if(i < n-1) printf(" "); // No trailing spaces
    }

    return 0;
}
```

Start Solving! (/editor/practice/6213)