

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И
ОПТИКИ**

Факультет программной инженерии и компьютерной техники

Кафедра «ВТ»

ЛАБОРАТОРНАЯ РАБОТА №_3_

ПО ДИСЦИПЛИНЕ «Система языкового программирования»

Выполнил(а): Чан Чунг Дык

Группа: P3202

Санкт-Петербург

2017/2018

9.1.16 Assignment: Scalar Product

The solution should consist of

- Two global arrays of int of the same size.
- A function to compute the scalar product of two given arrays.
- A main function which calls the product computations and outputs its results.

9.1.17 Assignment: Prime Number Checker

You have to write a function to test the number for primarity. The interesting thing is that the number will be of the type unsigned long and that it will be read from stdin.

- You have to write a function `int is_prime(unsigned long n)`, which checks whether n is a prime number or not. If it is the case, the function will return 1; otherwise 0.
- The main function will read an unsigned long number and call `is_prime` function on it. Then, depending on its result, it will output either yes or no.

Read man `scanf` and use `scanf` function with the format specifier `%lu`. Remember, `is_prime` accepts unsigned long, which is not the same thing as unsigned int!

Scalar Product	Prime Number Checker
<pre>#include <stdio.h> #define MAXN 20 int arr1[MAXN]; int arr2[MAXN]; int n; void input(){ scanf("%d",&n); for(int i=0; i < n*2 ; i++){ if (i<n) scanf("%d",&arr1[i]); else scanf("%d",&arr2[i-n]); } } long scalarPro(int* arr1, int* arr2){ long res=0; for(int i =0;i<n;i++){ res+= (long)arr1[i] * (long)arr2[i]; } return res; } int main(int argc, char** argv){ input(); printf("The scalar product is %d\n",scalarPro(arr1,arr2)); return 0; }</pre>	<pre>#include <stdio.h> unsigned long n ; void input(){ scanf("%lu",&n); } int is_prime(unsigned long n){ if (n<2) return 0; if (n<4) return 1; if (n%2==0 n%3==0) return 0; size_t i = 5; size_t w = 2; while (i<=n){ if (n%i == 0) return 0; i+=w; w= 6-w; } return 1; } int main(int argc, char** argv){ input(); if (is_prime(n) == 0) printf("NO"); else printf("YES"); return 0; }</pre>