НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Низкоуровневое программирование

Лабораторная работа 3

Выполнил
студент

Куприянов А.А

Группа P33113

г. Санкт-Петербург

2020

Скалярное произведение

Чтобы произвести скалярное произведение напишем функцию, которая принимает два указателя на массив (представляющий собой вектор):

```
int scalar(
        const int* vector1,
        const int* vector2,
        const size_t size)
{
    int result = 0;
    for (size_t i = 0; i < size; i++) {
        result += vector1[i] * vector2[i];
    }

    return result;
}
```

И вызовем этот метод для двух векторов (массивов):

```
int main() {
    const int a[] = {1,2,3};
    const int b[] = {666,666,666};

    int product = scalar(a, b, sizeof(a) / sizeof(a[0]));
    printf("Result is : %d", product);
    return 0;
}
```

Проверка на простоту

```c
int is_prime(unsigned long number) {

    if (number <= 1)
        return 0;

    for (unsigned long i = 2; i * i <= number; i++) {
        if (number % i == 0)
            return 0;
    }

    return 1;
}
```

Проверяем все числа до корня целевого числа

Также нужно считать ввод пользователя:

```c
int main() {
    unsigned long user_input;
    printf("Please, input your number: ");
    scanf("%lu", &user_input);

    printf("Your number [%lu] is %s",
            user_input,
            is_prime(user_input) ? "prime" : "not prime");

    return 0;
}
```

Вывод

В этой лабораторной работе были рассмотрены основные элементы языка программирования C. В основном условные конструкции, математические вычисления, а также типы данных.

Также в этой работе были использованы ключевые слова для препроцессора, например, для включения стандартного ввода-вывода:

```
#include<stdio.h>
```

Что позволяло использовть полезные функции вроде **printf** или **scanf**.