

**Университет ИТМО, кафедра ВТ**

**Лабораторная работа №3 по  
Языкам Системного Программирования**

Работу выполнил  
студент группы Р3200

**Рогов Я. С.**

Преподаватели:

**Жирков И.О.**

**Балакшин П.В.**

Санкт-Петербург, 2016

**Задание №1:** написать функцию sum, которая возвращать сумму массива из значений int.  
Содержимое файлов sum.c, libsum.h, libsum.c, inputnumber.c

#### sum.c

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

int array[] = {1,2,3,4,5,6,7,8,9,10};

int main( int argc, char** argv ) {
    printf("%d\n", sum(array, sizeof(array)));
    return 0;
}
```

#### libsum.h

```
#ifndef _SUM_H_
#define _SUM_H_

#include <stdio.h>
int sum(int[], size_t);
#endif /* _SUM_H_ */
```

#### libsum.c

```
#include "libsum.h"

int sum( int array[], size_t size_of_array){
    size_t i;
    const size_t size = size_of_array /
sizeof(array[0]) ;
    int result=0;
    for (i=0; i<size; i++)
        result+=array[i];
    return result;
}
```

#### inputnumber.c

```
#ifndef _INPUTNUMBER_C_
#define _INPUTNUMBER_C_ value

#include <stdio.h>
#include <stdlib.h>

static const char* const urbad = "Correct number,
please\n";

static char inputnumber(int* num){
    static char buffer[128];
    char * endp = buffer;
    char endc;

    while(1){
        endc = scanf("%s", buffer);
        *num = strtol(buffer, &endp, 10);

        /* if buffer parsed and ends
correctly */
        if(endp!=buffer && *endp==0)
            break;
        else if(EOF==endc)
            return endc;

        printf(urbad);
    }
    return endc;
}

#endif
```

**Задание №2:** написать функцию is\_prime, которая проверяла бы число на простоту.

#### prime.c

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

#include "inputnumber.c"

static const unsigned char g_byteprimes[] =
{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97,
101, 103, 107, 109, 113, 127, 131, 137, 139, 149,
151, 157, 163, 167, 173, 179, 181, 191, 193, 197,
199, 211, 223, 227, 229, 233, 239, 241, 251};

int is_prime(unsigned long);
static int dsqrt(unsigned long);
static int dlog2(unsigned long);

int main(void){
    int num;

    printf("Your number, please:\n");
    while(1){
        if (EOF==(inputnumber(&num))){
            printf(
                "No number passed\n");
            return 1;
        }
        if (num>=0){
            break ;
        }
        printf(
            "Not negative number, please\n");
    }
}
```

```
printf(is_prime(num) ? "yes\n" : "no\n");

return 0;
}

int is_prime(unsigned long num){
    int i;
    int bplength = sizeof(g_byteprimes)/
sizeof(*g_byteprimes);

    if (num<=g_byteprimes[bplength-1]){
        for (i=0; i<bplength; i++)
            if (num == g_byteprimes[i])
                return 1;

        return 0;
    }

    else{
        int bound;
        for (i=0; i<bplength; i++)
            if (!(num%g_byteprimes[i]))
                return 0;

        bound = dsqrt(num);
        for (i=g_byteprimes[bplength-1];
            i<=bound; i+=2)
            if (!(num%i))
                return 0;

        return 1;
    }
}
```

```

#define ITERATIONS 5

static int dsqrt(unsigned long num){
    int x,i;
    if (!num)
        return 0;
    if (num<3)
        return 1;
    x = num >> (dlog2(num)/2);
    for (i=0; i<ITERATIONS; i++)
        x = (x + num/x) /2;
    return x;
}

static int dlog2(unsigned long num){
    int count=0;
    do {
        num = num >> 1;
        count+=1;
    } while(num);
    return count;
}

```

**Задание №3:** реализовать программу, считывающую размер массива и его содержимое из стандартного потока ввода и выводит его сумму.

#### dsun.c

```

#include <stdio.h>
#include <stdlib.h>
#include "libsum.h"

#include "inputnumber.c"

static const char* const INPUTERROR = "Wrong input\n";
static const char* const MEMORYERROR = "Error during memory allocation\n";

int main(void){
    int size;
    int * numbers;
    int i;

    printf("Array size, please: \n");
    if(EOF==(inputnumber(&size)) || size<0){
        printf(INPUTERROR);
        return 1;
    }

    numbers = (int* ) malloc(sizeof(int) * size);

    printf("Your numbers, please: \n");
    for(i=0; i<size; i++){
        if(EOF==(inputnumber(numbers+i))){
            printf(INPUTERROR);
            return 1;
        }

        printf("%d\n", sum(numbers, size * sizeof(numbers[0])));
        free(numbers);
        return 0;
    }
}

```

**Вывод:** в ходе выполнения данной лабораторной работы я ознакомился с основами работы с массивами и указателями в С, некоторыми функциями работы с вводом/выводом, а также способами динамического выделения памяти.