|  |  |
| --- | --- |
| C:\Documents and Settings\admin\Рабочий стол\МИРЭА_ЭМБЛЕМА_приказ.JPG | |
| МИНОБРАНАУКИ РОССИИ | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  «**МОСКОВСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»**  **МИРЭА** | |
| Институт информационных технологий  Кафедра математического обеспечения и стандартизации информационных технологий | |
| **Отчет по лабораторной работе №5**  **Тема работы «ФУНКЦИИ»** | |
| Выполнил: |  |
| Студент группы ИКБО-07-17  Отчет принял:  Преподаватель кафедры МОСИТ | Акжигитов Р. Р.  Миронов Антон Николаевич |
| Москва  2018 | |

Оглавление

[Задание 1 3](#_Toc510983017)

[Условия задания 3](#_Toc510983018)

[Декомпозиция 3](#_Toc510983019)

[Реализация функций 3](#_Toc510983020)

[Задание 2 3](#_Toc510983021)

[Условия задания 3](#_Toc510983022)

[Декомпозиция 3](#_Toc510983023)

[Реализация функций 3](#_Toc510983024)

[Функции ввода\вывода массива\значений 4](#_Toc510983025)

[Кодирование алгоритма программы 4](#_Toc510983026)

[Вывод 6](#_Toc510983027)

# Задание 1

# Условия задания

1. 1.Разработать функцию, которая вычисляет количество единиц в двоичном коде целого числа.
2. Разработать процедуру, которая в последовательности целых чисел находит число, в двоичном представлении которого больше всего единиц и количество единиц в этом числе.

# Декомпозиция

* Используя сдвиг битов, посчитать кол-во четных чисел => кол-во единиц в 2-ой записи числа.
* Считать кол-во единиц в каждом числе, искать число с максимальным кол-вом единиц, запомнить само число и кол-во единиц.

# Реализация функций

// кол-во единиц в двоичной записи числа

int amount1(int n) {

int cx = 0;

for (; n > 0; cx = (n % 2) ? cx + 1 : cx, n >>= 1);

return cx;

}

// получение числа, в дв. представлении которого больше всего единиц, и кол-ва единиц в этом числе

void getMaxAmount(int amount, int &number, int &amountOf1) {

int num = 0;

for (int i = 0; i < amount; i++) {

input(num);

int am = amount1(num);

amountOf1 = amountOf1 < am ? number = num, am : amountOf1;

}

}

# Задание 2

# Условия задания

1. Дана последовательность из n целых чисел. Определить количество чисел, содержащих в своей десятичной записи хотя бы одну цифру 7 и являющихся простыми числами.

# Декомпозиция

* Функция проверки на простоту, цикл for с проверкой, если хотя бы раз поделилось без остатка возврат из функции false, если ни разу, то true.
* Содержит ли в записи число, делим на 10 и проверяем остаток деления на 10, если он равен 7, то возвращаем true.
* Функция для подсчета кол-ва чисел, удовлетворяющих обоим условиям.

# Реализация функций

// содержит ли число num хотя бы одну цифру div

int consistNumber(int num, int div) {

for (; num > 0; num /= 10)

if (num % 10 == div)

return 1;

return 0;

}

// простое ли число

int isPrime(int n) {

if (n < 2) return 0;

for (int i = 2; i \* i <= n; i++) {

if (!(n % i)) return 0;

}

return 1;

}

// кол-во чисел одновременно содержащих 7 и простых

int seq(int n) {

int cx = 0, num = 0;

for (int i = 0; i < n; i++) {

cin >> num;

cx = (consistNumber(num, 7) && isPrime(num)) ? cx + 1 : cx;

}

return cx;

}

# Функции ввода\вывода массива\значений

// контроль вводимых значений

int input(int &number, int min = INT\_FAST16\_MIN, int max = INT\_FAST16\_MAX) {

while (!(cin >> number) || (number < min || number > max)) {

cout << "\nPlease input integer number [" << min << ".." << max << "]... Your choice's: ";

}

return number;

}

# Кодирование алгоритма программы

// laba5.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include "iostream"

using namespace std;

// контроль вводимых значений

int input(int &number, int min = INT\_FAST16\_MIN, int max = INT\_FAST16\_MAX) {

while (!(cin >> number) || (number < min || number > max)) {

cout << "\nPlease input integer number [" << min << ".." << max << "]... Your choice's: ";

}

return number;

}

// кол-во единиц в двоичной записи числа

int amount1(int n) {

int cx = 0;

for (; n > 0; cx = (n % 2) ? cx + 1 : cx, n >>= 1);

return cx;

}

// получение числа, в дв. представлении которого больше всего единиц, и кол-ва единиц в этом числе

void getMaxAmount(int amount, int &number, int &amountOf1) {

int num = 0;

for (int i = 0; i < amount; i++) {

input(num);

int am = amount1(num);

amountOf1 = amountOf1 < am ? number = num, am : amountOf1;

}

}

// содержит ли число num хотя бы одну цифру div

int consistNumber(int num, int div) {

for (; num > 0; num /= 10)

if (num % 10 == div)

return 1;

return 0;

}

// простое ли число

int isPrime(int n) {

if (n < 2) return 0;

for (int i = 2; i \* i <= n; i++) {

if (!(n % i)) return 0;

}

return 1;

}

// кол-во чисел одновременно содержащих 7 и простых

int seq(int n) {

int cx = 0, num = 0;

for (int i = 0; i < n; i++) {

cin >> num;

cx = (consistNumber(num, 7) && isPrime(num)) ? cx + 1 : cx;

}

return cx;

}

int main()

{

cout << "IKBO-07-17 Akzhigitov Radmir (Lab. 5, I variant) 06.03.18\n";

int flag = 0;

while (flag != 4) {

cout << "1) Get amount of 1's from number\n2) Get number and amount where amount of 1's is max\n3) Get amount of prime and consisting 7 numbers\n4) Exit\n";

input(flag, 1, 4);

int inputted;

switch (flag) {

case 1: {

cout << "Enter integer: ";

cout << "Amount of 1's: " << amount1(input(inputted)) << "\n\n";

system("pause");

system("cls");

break;

}

case 2: {

int n, maxNum, maxAmount;

cout << "Enter amount of integers: ";

cin >> n;

getMaxAmount(n, maxNum, maxAmount);

cout << maxNum << " consists " << maxAmount << " 1's\n\n";

system("pause");

system("cls");

break;

}

case 3: {

int n;

cout << "Enter amount of integers: ";

cin >> n;

cout << "Enter " << n << " integers: ";

cout << "Amount of numbers: " << seq(n) << "\n\n";

system("pause");

system("cls");

break;

}

case 4: break;

}

}

system("pause");

return 0;

}

# Вывод

Выполнив работу, я узнал, как можно объявлять, инициализировать функции, какие существуют механизмы передачи параметров, понял, зачем нужны прототипы, как вызывать функции из подпрограмм.