ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ

За курсом «Інформатика і Програмування»

Студента групи ПА-19-1

Макаров Олег Ігорович

Факультет прикладної математики

Кафедра Комп'ютерне моделювання та технології програмування

1. **Постановка задачі**

**Составить программу SUM**, которая:

==========

1) вводит с клавиатуры последовательность целых чисел,

заканчивающуюся нулем;

2) находит сумму всех членов последовательности;

3) выводит на экран найденное значение в виде:

"SUM= ...".

Пример

Введите последовательность целых чисел. Ноль - признак конца.

a[1]= 13

a[2]= 21

a[3]= 2

a[4]= 14

a[5]= -1

a[6]= -2

a[7]= 0

SUM=47

1. **Опис розв’язку**

Послідовність вводиться користувачем, доки користувач не введе нуль, який позначає кінець послідовності, потім виводиться сума елементів послідовності.

1. **Вихідний текст програми розв’язку задачі**

#include<iostream>

using namespace std;

int main()

{

int a, SUM = 0, i = 1;

cout << "\na[" << i << "]= ";

while (cin >> a)

{

if (a != 0)

{

i++;

SUM += a;

cout << "\na[" << i << "]= ";

}

else break;

}

cout << "\nSUM= " << SUM;

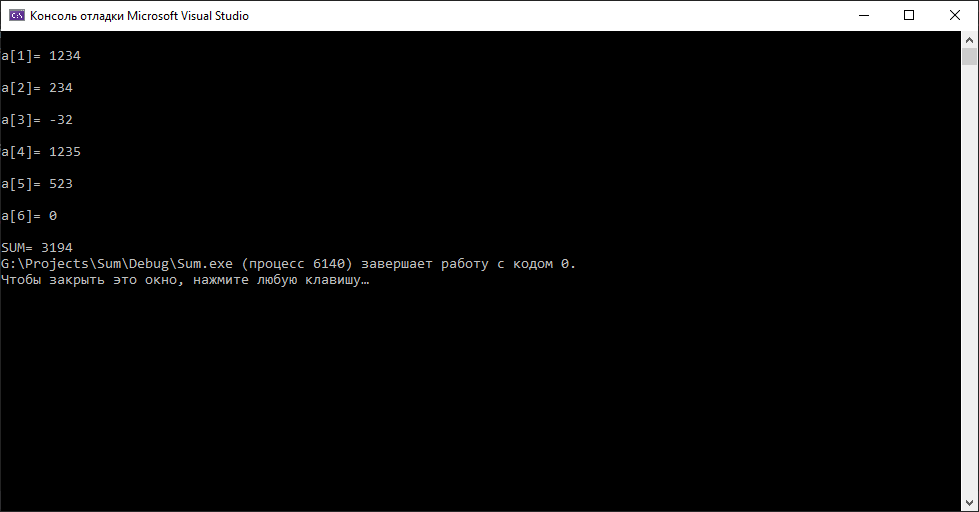
return 0;

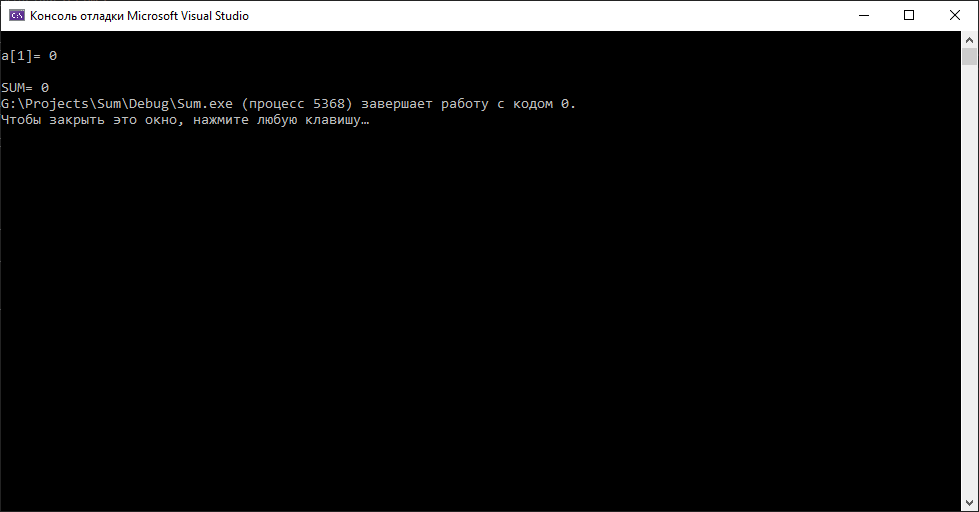
}

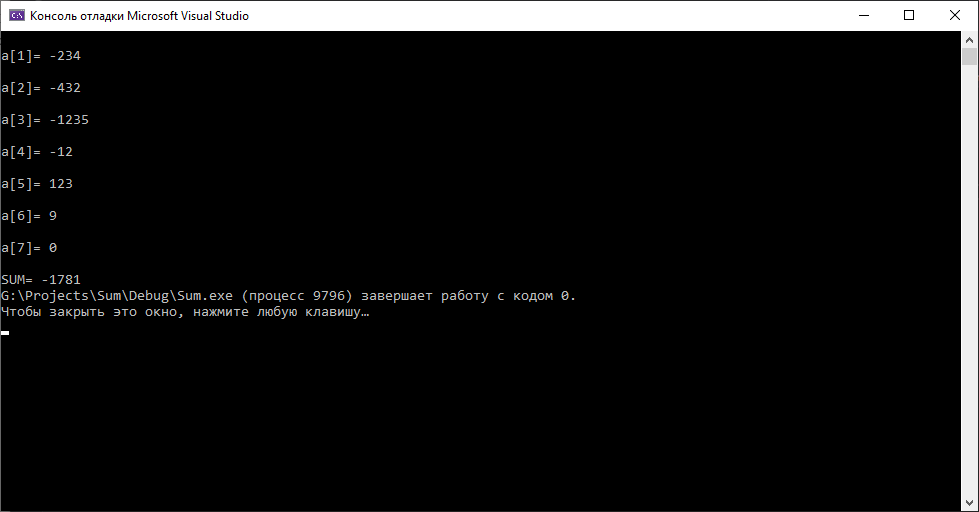
1. **Опис інтерфейсу (керівництво користувача)**

Після запуску програми користувач вводить послідовність, окремо кожне число. Після виведення суми послідовності користувач має натиснути будь-яку клавішу, щоб зачинити програму.

1. **Опис тестових приладів**







1. **Аналіз помилок(опис усунення зауважень)**
2. **Постановка задачі**

1) вводит с клавиатуры последовательность целых чисел,

заканчивающуюся нулем;

2) находит произведение всех четных членов последовательности;

3) выводит на экран вычисленное произведение в виде:

"ПРОИЗВЕДЕНИЕ= ...".

Пример

Введите последовательность целых чисел. Ноль - признак конца.

a[1]= 13

a[2]= 21

a[3]= 2

a[4]= 14

a[5]= -1

a[6]= -2

a[7]= 0

ПРОИЗВЕДЕНИЕ=-56

1. **Опис розв’язку**

Послідовність вводиться користувачем, доки користувач не введе нуль, який позначає кінець послідовності, потім виводиться добуток парних елементів послідовності.

1. **Вихідний текст програми розв’язку задачі**

#include<iostream>

using namespace std;

int main()

{

int a, dob = 1, i = 1;

cout << "\na[" << i << "]= ";

while (cin >> a) {

if (a != 0) {

if (a % 2 == 0) {

dob \*= a;

}

i++;

cout << "\na[" << i << "]= ";

}

else break;

}

cout << "\nDOBUTOK= " << dob;

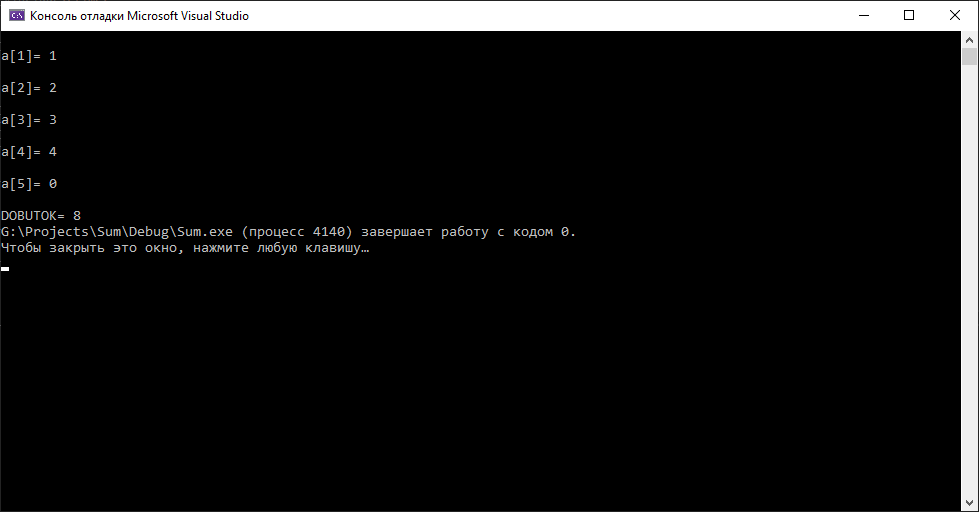
return 0;

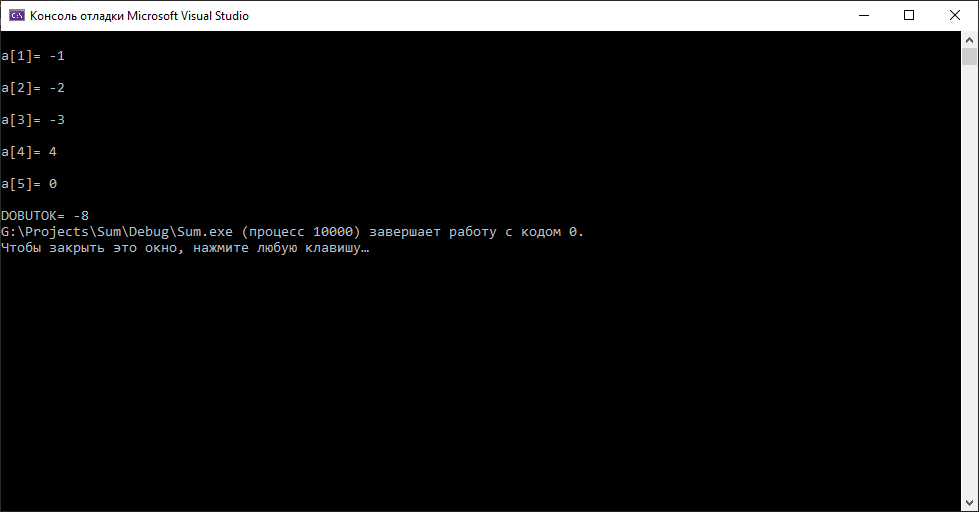
}

1. **Опис іинтерфейсу (керівництво користувача)**

Після запуску програми користувач вводить послідовність, окремо кожне число. Після виведення добутку парних чисел послідовності користувач має натиснути будь-яку клавішу, щоб зачинити програму.

1. **Опис тестових прикладів**





1. **Аналіз помилок (опис усунення зауважень)**
2. **Постановка задачі**

Составить программу CountOdd, которая:

1. Вводит с клавиатуры последовательность целых чисел.

2. Считает сколько среди них нечетных.

3. Как только вводится число 0, программа прекращает работу

и выдает результат подсчета на экран.

Пример работы программы:

Введите число= 11

Введите число= 10

Введите число= 3

Введите число= 1

Введите число= 2

Введите число= 0

Встретилось 3 нечетных числа.

1. **Опис розв’язку**

Послідовність вводиться користувачем, доки користувач не введе нуль, який позначає кінець послідовності, потім виводиться кількість непарних елементів послідовності.

1. **Вихідний текст програми розв’язку задачі**

#include<iostream>

using namespace std;

int main()

{

int a, odd = 0, i = 1;

cout << "\na[" << i << "]= ";

while (cin >> a) {

if (a != 0) {

if (a % 2 != 0) {

odd++;

}

i++;

cout << "\na[" << i << "]= ";

}

else break;

}

cout << "\nKol-vo odd chisel= " << odd;

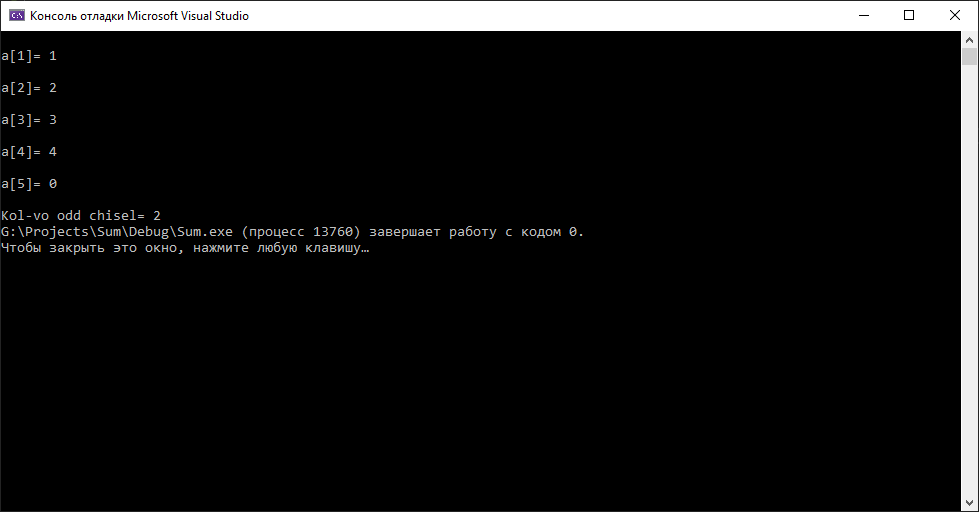
return 0;

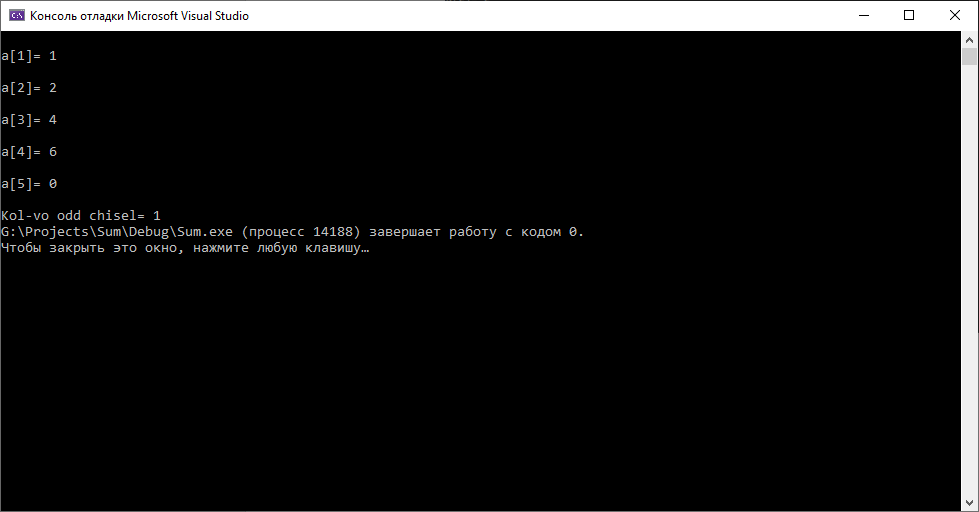
}

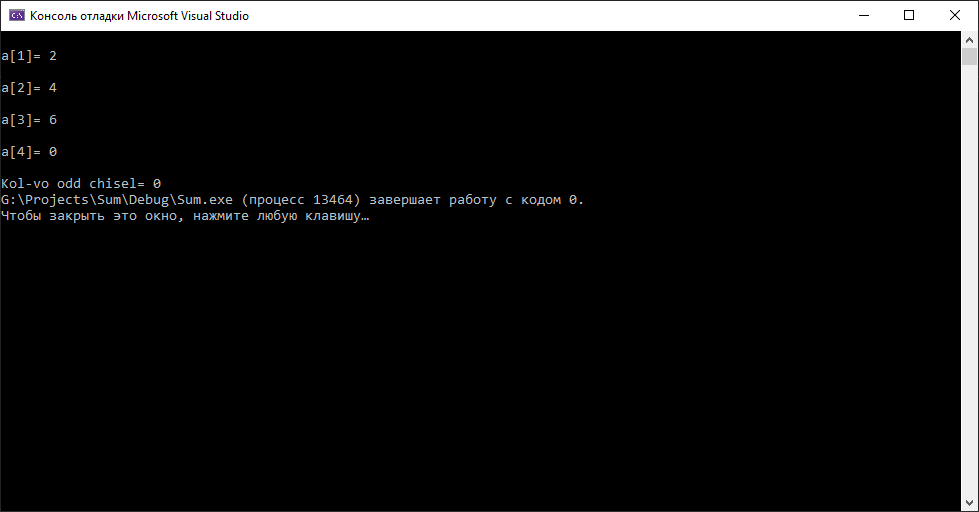
1. **Опис іинтерфейсу (керівництво користувача)**

Після запуску програми користувач вводить послідовність, окремо кожне число. Після виведення кількості непарних чисел послідовності користувач має натиснути будь-яку клавішу, щоб зачинити програму.

1. **Опис тестових прикладів**







1. **Аналіз помилок (опис усунення зауважень)**
2. **Постановка задачі**

Составьте программу PARROT2, которая:

1. Просит пользователя ввести с клавиатуры

одну строку какого-то текста.

2. Выводит эту строку на экран.

3. Повторяет эти действия до тех пор,

пока пользователь не введёт пустую строку

(то есть не нажмёт просто ENTER).

1. **Опис розв’язку**

Користувач вводить строки тексту, доки не введе пусту строку. А програма виводить цей текст на екран.

1. **Вихідний текст програми розв’язку задачі**

#include<iostream>

using namespace std;

int main()

{

string str = "qwe";

cout << "Enter any text(English only): ";

getline(cin, str);

while (1) {

getline(cin, str);

if (str == "\0")

break;

cout << "\n" << str << endl;

cout << "\nEnter any text: ";

}

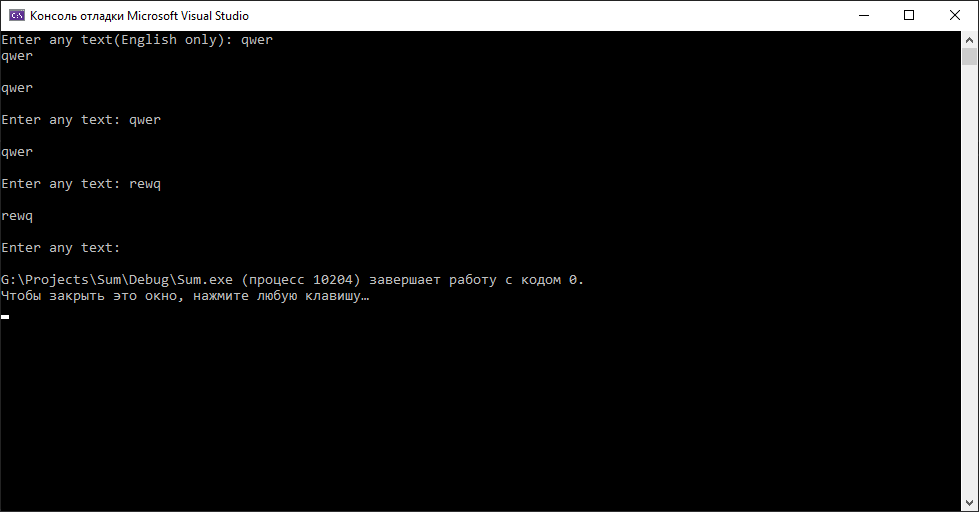
return 0;

}

1. **Опис іинтерфейсу (керівництво користувача)**

Після запуску програми користувач вводить текси, окремо кожну строку. Після виведення тексту користувач має натиснути ENTER, щоб зачинити програму.

1. **Опис тестових прикладів**



1. **Аналіз помилок (опис усунення зауважень)**
2. **Постановка задачі**

1) вводит с клавиатуры последовательность целых чисел,

заканчивающуюся нулем;

2) находит наибольший из всех членов последовательности;

3) выводит на экран найденное значение в виде:

"MAX= ...".

Пример

Введите последовательность целых чисел. Ноль - признак конца.

a[1]= 13

a[2]= 21

a[3]= 2

a[4]= 14

a[5]= -1

a[6]= -2

a[7]= 0

MAX=21

1. **Опис розв’язку**

Послідовність вводиться користувачем, доки користувач не введе нуль, який позначає кінець послідовності, потім виводиться максимальний елемент послідовності.

1. **Вихідний текст програми розв’язку задачі**

#include<iostream>

#include<string>

#include<ctime>

using namespace std;

int main()

{

int a, MAX = 0, i = 1;

cout << "\na[" << i << "]= ";

cin >> a;

for (i = 1; a != 0; cin >> a) {

if (i == 1)

MAX = a;

if (a > MAX)

MAX = a;

i++;

cout << "\na[" << i << "]= ";

}

cout << "\nMAX= " << MAX;

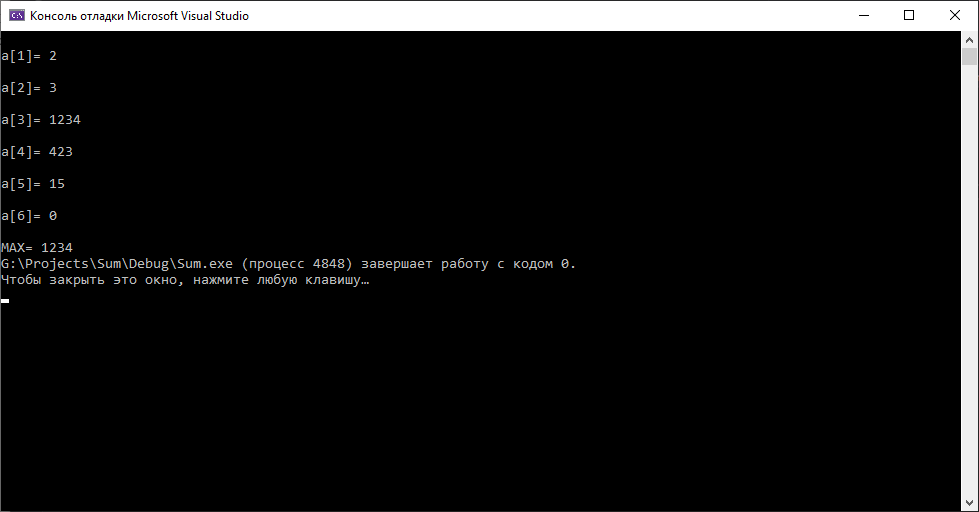
return 0;

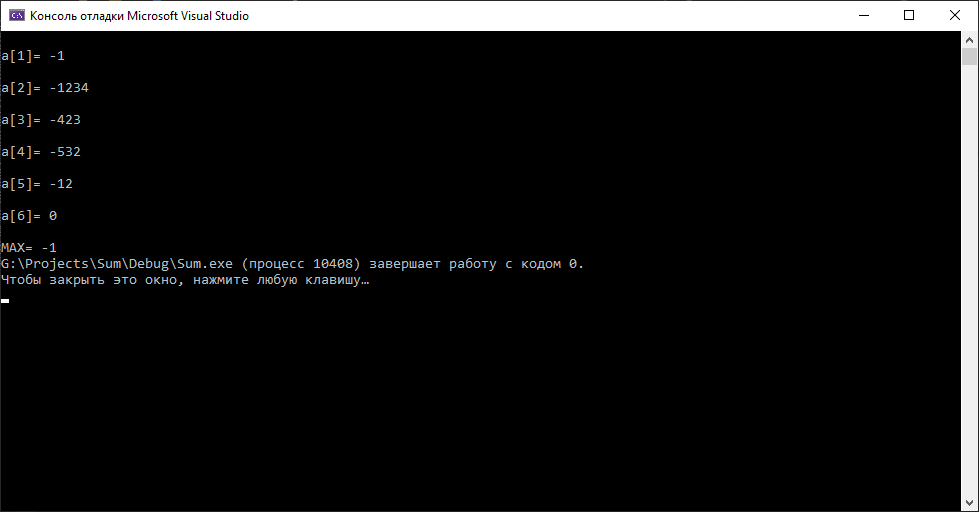
}

1. **Опис іинтерфейсу (керівництво користувача)**

Після запуску програми користувач вводить послідовність, окремо кожне число. Після виведення максимального елемента послідовності користувач має натиснути будь-яку клавішу, щоб зачинити програму.

1. **Опис тестових прикладів**





1. **Аналіз помилок (опис усунення зауважень)**