Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики

и радиоэлектроники»

Специальность «Программная инженерия»

Кафедра инженерной психологии и эргономики

Учебная дисциплина «Основы Алгоритмизации и программирования»

Отчет

по лабораторной работе №8

«Функции»

Вариант 18

Подготовил: Козинцев М.Л

Проверил: Усенко Ф.В.

Минск 2024

***Цель работы: с***формировать навыки и умения обработки структурированных типов данных, организованных в виде функций.

***Задание 1:*** Дано натуральное число P. Найдите сумму цифр числа P.

#include <iostream>

using namespace std;

int how\_much\_digits(int num);

int digits\_summa(int num);

int main() {

int a;

cout << "Enter a number: ";

cin >> a; //Ввод числа

cout << "Summa of " << a << " digits is: " << digits\_summa(a) << '\n';

return 0;

}

int how\_much\_digits(int num) { //Функция для подсчёта количества цифр в числе

int loc = num, counter = 1;

while (loc / 10 != 0) {

counter++;

loc /= 10;

}

return counter;

}

int digits\_summa(int num) { //Функция для подсчёта суммы цифр числа

int loc = num, divider, length = how\_much\_digits(num), summa = 0;

for (int i = length; i > 0; i--) {

divider = pow(10, i - 1);

summa += loc / divider;

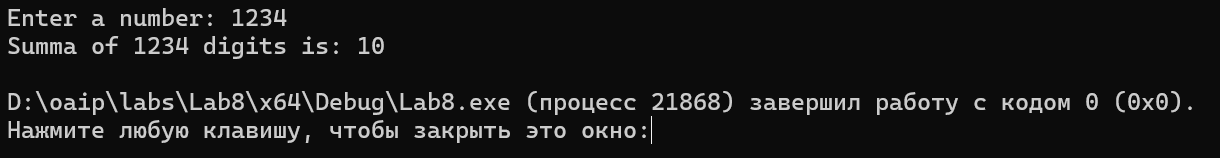
loc -= loc / divider \* divider;

}

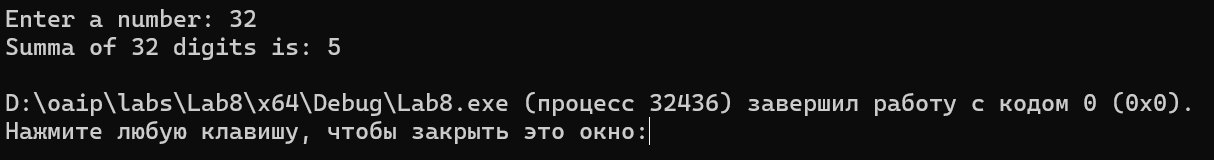
return summa;

}

Результаты работы программы представлены на рисунках 1-2.

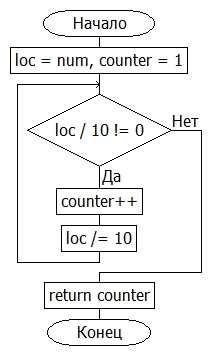


(Рис. 1)

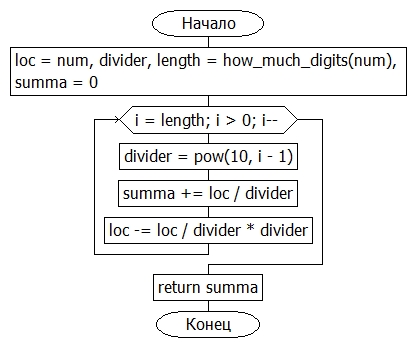


(Рис. 2)

Блок-схема функции для определения количества цифр числа (how\_much\_digits) представлена на рисунке 3.

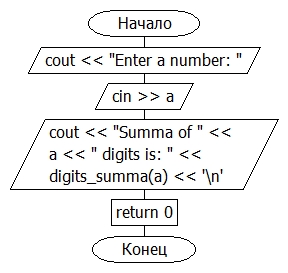


(Рис.3)

Блок-схема функции нахождения суммы цифр числа (digits\_summa) представлена на рисунке 4.

(Рис. 4)

Блок-схема работы программы представлена на рисунке 5.



(Рис. 5)

***Задание 2:*** Используя перегрузку методов, создайте программу согласно варианту.

а) для последовательности целых чисел находит максимальный элемент;

б) для строки находит длину самого длинного слова.

#include <iostream>

#include <string>;

using namespace std;

double max\_of(double\* numbers, int length);

int max\_of(string str);

int main() {

double \*arr;

int choice, leng, i;

string str;

char ignore;

cout << "Choose what to enter:\n1 - for numbers\n2 - for words\n";

cin >> choice;

switch (choice) { //Запрос на ввод строки или последовательности

case 1:

cout << "Enter length of number array: ";

cin >> leng;

arr = new double[leng];

cout << "Enter elements of array";

for (i = 0; i < leng; i++) { //Заполнение последовательности

cin >> arr[i];

}

cout << "Max from number array is: " << max\_of(arr, leng);

delete[] arr;

break;

case 2:

cout << "Enter your line: ";

cin.ignore();

getline(cin, str); //Ввод строки

cout << "\nMax words length is: " << max\_of(str);

break;

}

return 0;

}

int max\_of(string str) { //Нахождение длины самого длинного слова в строке

int length = 0, max\_length = 0;

for (int i = 0; i < str.size(); i++) {

if (str[i] != ' ') {

length++;

}

else if (str[i] == ' ' && length > max\_length) {

max\_length = length;

length = 0;

}

else {

length = 0;

}

}

if (length > max\_length) {

max\_length = length;

}

return max\_length;

}

double max\_of(double\* numbers, int length) { //Нахождение максимального элемента последовательности

double maxx = numbers[0];

int i;

for (i = 0; i < length; i++) {

if (numbers[i] > maxx) {

maxx = numbers[i];

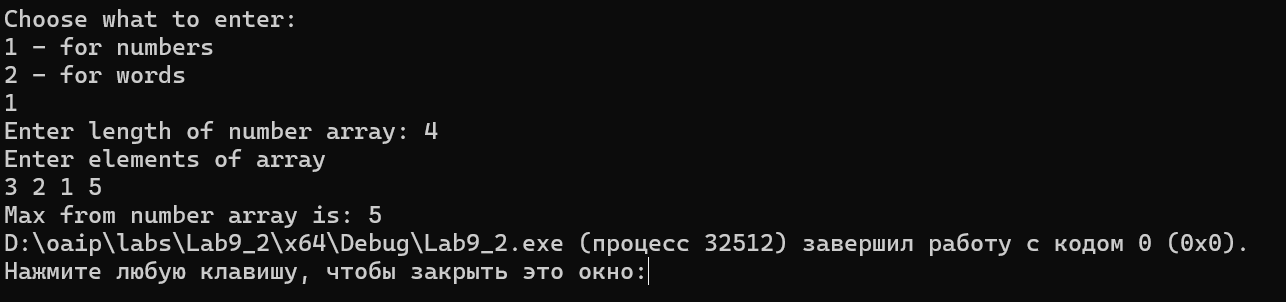
}

}

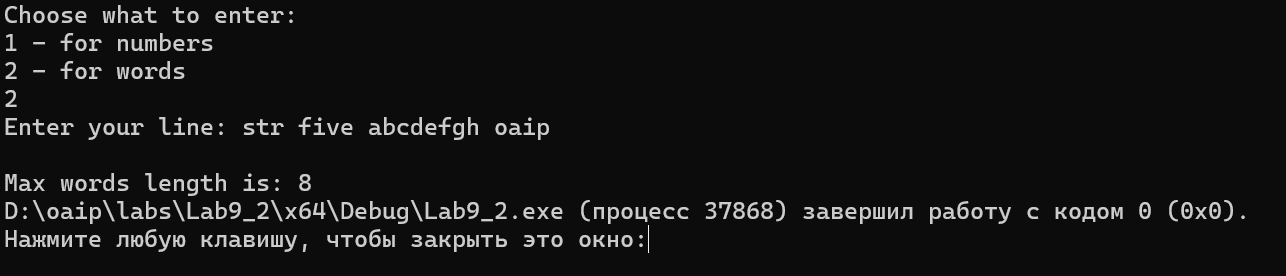
return maxx;

}

Результаты работы программы представлены на рисунках 6-7.



(Рис. 6)

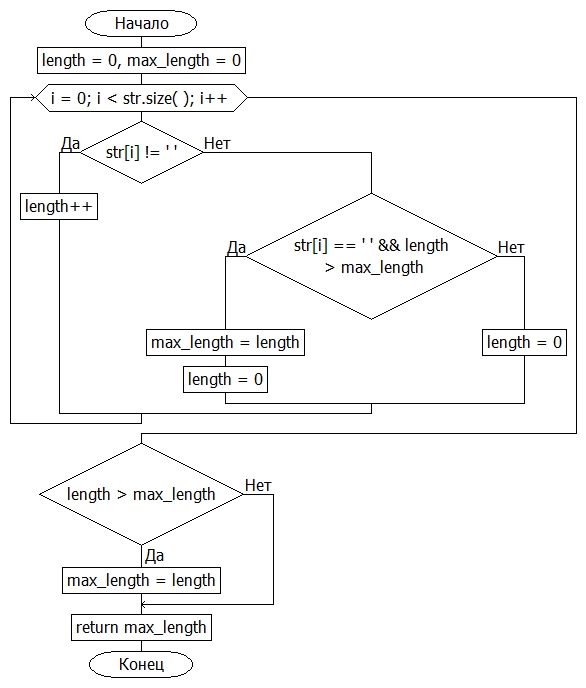
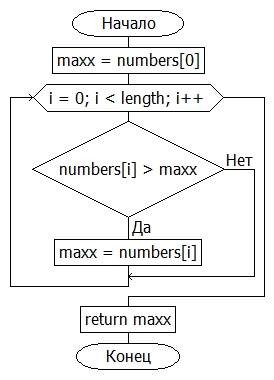


(Рис. 7)

Блок-схема функции нахождения максимального числа последовательности (max\_of) представлена на рисунке 8.

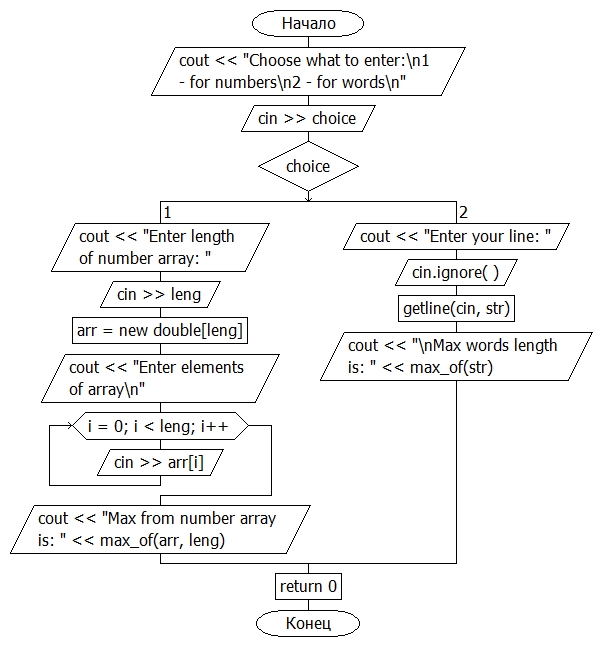
(Рис. 8)

Блок-схема функции нахождения максимальной длины слова в строке (max\_of) представлена на рисунке 9.



(Рис. 9)

Блок-схема работы программы представлена на рисунке 10.



(Рис. 10)

***Вывод***

В ходе выполнения работы была достигнута цель данной лабораторной работы: ***с***формировать навыки и умения обработки структурированных типов данных, организованных в виде функций.