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 No4

«Функции»

Вариант 17

Подготовил: Ковальчук А.А.

Проверил: Усенко Ф.В.

Минск 2024

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

Задание:

1. Дано натуральное число P. Найдите все делители числа P.
2. а) для последовательности целых чисел находит среднее арифметическое;

б) для строки находит количество букв, содержащихся в ней.

#include <iostream>

#include <string>

using namespace std;

// Функция для нахождения и вывода всех делителей числа p

void devision(int p) {

for (int i = 1; i <= p; i++) {

if (p % i == 0) { // Проверка, является ли i делителем p

cout << i << endl; // Вывод делителя

}

}

}

int main() {

int p; // Переменная для хранения введенного числа

cout << "Enter p: ";

cin >> p; // Ввод числа пользователем

devision(p); // Вызов функции для нахождения и вывода делителей

return 0;

}

#include <iostream>

#include <string>

#include <sstream>

#include <cctype>

using namespace std;

void calculate(string str);

void calculate(string str, int n);

int main() {

setlocale(LC\_ALL, "RU");

string str; // Переменная для хранения введенной строки

cout << "Введите строку: "; // Вывод приглашения для ввода строки

while (getline(cin, str)) { // Проверка на пустую строку

if (!str.empty()) { // Если строка не пустая, выйти из цикла

break;

}

cout << "Пожалуйста, введите строку с некоторыми символами: "; // Запрос на ввод непустой строки

}

// Подсчет количества слов в строке (количество пробелов + 1)

int n = count(str.begin(), str.end(), ' ') + 1;

// Проверка каждого символа в строке

for (char c : str) {

// Если символ не является цифрой, знаком '-' или пробелом

if (!isdigit(c) && c != '-' && c != ' ') {

calculate(str); // Вызов функции для обработки строк с буквами

return 0;

}

}

// Если строка содержит только цифры

calculate(str, n); // Вызов функции для обработки строк с числами

return 0;

}

// Функция для подсчета букв в строке

void calculate(string str) {

int count = 0; // Счетчик букв

for (char c : str) {

if (isalpha(c)) { // Проверка, является ли символ буквой

count++; // Увеличение счетчика

}

}

cout << "Было найдено " << count << " букв" << endl; // Вывод результата

}

// Функция для подсчета среднего арифметического чисел в строке

void calculate(string str, int n) {

double sum = 0; // Сумма чисел

int count = 0; // Счетчик чисел

istringstream stream(str); // Создание потока из строки

string numStr; // Переменная для хранения числа в виде строки

// Извлечение чисел из строки

while (stream >> numStr) {

sum += stod(numStr); // Преобразование строки в число и добавление к сумме

count++; // Увеличение счетчика чисел

}

double res = sum / count; // Вычисление среднего арифметического

cout << "Среднее арифметическое равно: " << res << endl; // Вывод результата

}

Результат работы программы представлен на рисунке 1.

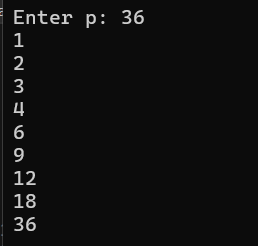


Рисунок 1

Результат работы программы представлен на рисунке 2.



Рисунок 2

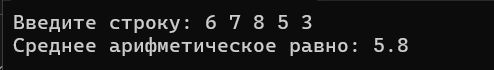


Рисунок 2

Блок-схема работы программы представлена на рисунке 3.

devision

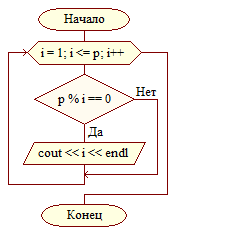


Рисунок 3

main

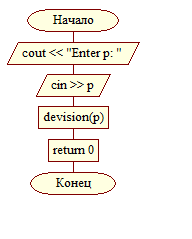


Рисунок 3

Блок-схема работы программы представлена на рисунке 4.

calculate(string str)

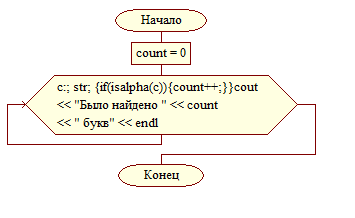


Рисунок 4

calculate(string str, int n)

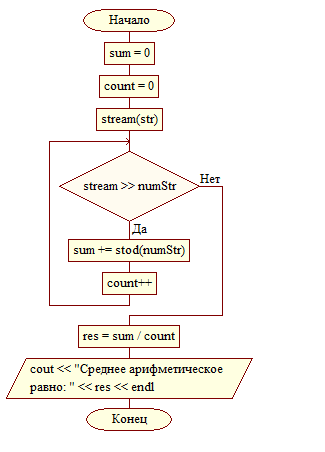


Рисунок 4

main

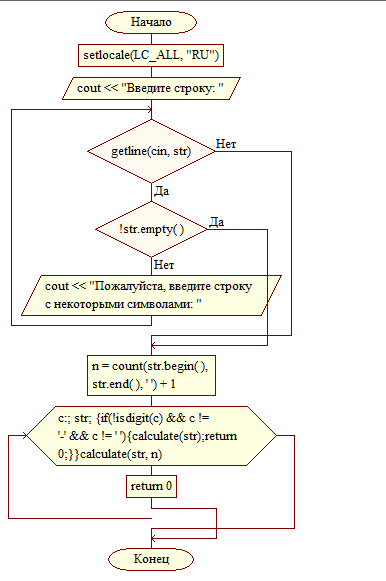


Рисунок 4

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