**Вариант 5**

1. **Строки (часть 1, лаб 6, 6.4.2)**

В строке, состоящей из групп нулей и единиц, подсчитать количество единиц в группах с нечетным количеством символов.

#include <iostream>

#include <string>

#include <stdlib.h>

int main() {

using namespace std;

string s;

int count1 = 0, count2 = 0; // 1 - счетчик единиц в группе, 2 - счетчик единиц по всем группам

cout << "Enter the string: " << endl;

getline(cin, s);

for (int i = 0; i < s.length(); i++) {

if (s[i] == '1') {

count1++;

}

else if (s[i] == '0') {

if (count1 % 2) {

count2 += count1;

}

count1 = 0;

}

else { //если элемент строки не является 0 или 1, выводим сообщение об ошибке ввода

cout << "input error" << endl;

exit(0);

}

}

if (count2 != 0) {

cout << count2<<endl;

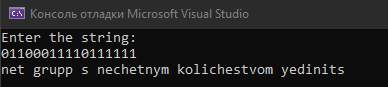
}

else cout << "net grupp s nechetnym kolichestvom edinits" << endl;

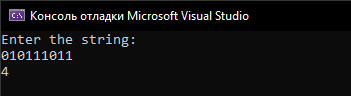
return 0;

}

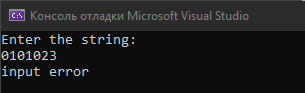
1. Когда во всех группах с единицами содержится четное количество единиц



1. Когда есть группы с нечетным количеством единиц



1. Когда в введенной строке есть символы, не равные «0» или «1».



1. **Файлы (часть 1, лаб 7.4.1)**

Написать программу обработки файла записей, содержащую следующие подпункты меню: «Создание», «Просмотр», «Добавление», «Решение индивидуального задания». Каждая запись должна содержать следующую информацию о студентах:

- фамилия

- номер группы

- оценки за семестр: по физике, математике и информатике

- средний балл

Организовать ввод исходных данных, средний балл рассчитать по введенным оценкам. Содержимое всего файла и результаты решения индивидуального задания записать в текстовый файл.

Задание:

Найти информацию о студентах, имеющих отметку 4 или 5 по физике и больше 8 по остальным предметам.

1. **Рекурсия (часть 2, лаб 1, 1.3, стр 7)**

Найти значение функции Аккермана  *A(m, n),* которое определяется для всех неотрицательных целых аргументов m и n следующим образом:

A(O, n) = n + 1

A(m, O) = A(m-1, 1) при m > 0

A(m, n) = A(m-1, A(m, n-1)) при m > 0 и n > 0

#include <iostream>

using namespace std;

int A(int m, int n) {

if (m == 0) return n + 1;

else {

if ((m > 0) && (n == 0)) {

return A(m - 1, 1);

}

else if ((m > 0) && (n > 0)) {

return A(m - 1, A(m, n - 1));

}

}

}

int main() {

int m = 0, n = 0;

cout << "Enter m: " << endl;

cin >> m;

cout << "Enter n: " << endl;

cin >> n;

cout << "The result: " << A(m, n) << endl;

return 0;

}

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Работа программы    Проверка введенных данных на тип и отрицательность | Результат выполнения программы   |  |  |  | | --- | --- | --- | | **m** | **n** | **result** | | 0 | 0 | 1 | | 0 | 1 | 2 | | 0 | 2 | 3 | | 0 | 3 | 4 | | 1 | 0 | 2 | | 1 | 1 | 3 | | 1 | 2 | 4 | | 1 | 3 | 5 | | 2 | 0 | 3 | | 2 | 1 | 5 | | 2 | 2 | 7 | | 2 | 3 | 9 | | 3 | 0 | 5 | | 3 | 1 | 13 | | 3 | 2 | 29 | | 3 | 3 | 61 | | 4 | 0 | 13 | |

1. **Стеки (часть 2, лаб 3, 3.3, стр 20)**

Написать программу по созданию, добавление, просмотру и решению приведенных дальше задач для однонаправленного линейного списка типа Stack. Реализовать сортировку стека методами, рассмотренными в подразделе 3.1.

Из созданного списка удалить элементы, заканчивающиеся на цифру 5

1. **Очереди (часть 2, лаб 4, стр 26)**

Написать программу по созданию, добавлению (в начало, в конец), просмотру (с начала, с конца) и решению приведенной в подразделе 3.3.задачи для двунаправленных линейных списков

Из созданного списка удалить элементы, заканчивающиеся на цифру 5

1. **Нелинейные списки (часть 2, лаб 6, стр 40)**

Определить число узлов на каждом уровне дерева.

1. **Курсач**

Написать программу обработки файла данных, состоящих из структур, в которой реализованы следующие функции:

Стандартная обработка файла( создание, просмотр, добавление записи)

- Сортировка массива методами прямого выбора

+ быстрая сортировка ДОДЕЛАТЬ

- Двоичный поиск в отсортированном массиве

+Линейный поиск в файле ДОДЕЛАТЬ

Список разговоров на международной АТС

Вид списка: дата разговора, код и название города, время разговора, тариф, номер телефона в этом городе и номер телефона абонента

Здание: вывести по каждому городу общее время разговоров с ним и сумму.