Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Инженерно-технологическая академия ЮФУ

Институт компьютерных технологий и информационной безопасности

**Кафедра Математическое обеспечение и администрирование информационных систем Физико- математических основ и основ инженерного образования**

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

по курсу: «Алгоритмизация и программирование»

Выполнил

студент группы КТбо 1-1

№ КТ-02.03.03.01-о1 А. В. Бураков

№ зачетки

Проверил

Ст. преподаватель кафедры ВТ Д.С. Кочубей

«27» декабря 2023

Таганрог, 2023

# Техническое задание

## Цель работы

Задание состоит в разработке программы, которая считывает настроечные параметры и формирует отчет по имеющимся данным в файлах (см. Таблица 1). Отчет записывается в текстовый файл. Число записей в каждом из описанных выше файлов произвольно.

Таблица 1 – Задание варианта 7

|  |  |  |
| --- | --- | --- |
| № варианта | Отчёт | Параметры |
| 5 | Определить список клиентов и список услуг, которыми они пользовались за указанный месяц | Номер месяца |

**Описание задания**

Имеется информация о клиентах телефонной компании и предоставляемых им услугах. Каждая услуга имеет собственный тариф, а каждый клиент может пользоваться произвольным набором услуг в течение ограниченного интервала времени (соответственно срокам договора). Имеются данные о фактическом использовании услуг.

Вся указанная информация представлена текстовыми файлами, структура которых выглядит следующим образом.

Файл информации о клиентах содержит фамилию, имя, отчество клиента, номер телефона, дату заключения договора, дату окончания договора, размер задолженности, допустимый кредит. Каждое поле отделяется запятой, запись – это строка текста. Пример содержимого в файле:

Иванов Иван Иванович, 9773672365, 12.10.2012, 12.10.2014,0,0

Петров Иван Васильевич, 9734672311, 22.01.2008, 01.11.2011,210,200

Васильев Илья Васильевич, 9714679805, 05.09.2010, 01.12.2012,0,2000

Файл информации об услугах состоит также из записей, состоящих из полей, разделенных запятыми. Каждая запись включает наименование услуги, ее код, тариф (в рублях), временной интервал измерения (мин., сутки, месяц. Если временной привязки нет, ставится символ #). Пример данных в файле:

Cвязь внутри сети, 1, 0.30, мин

СМС, 2, 0.15, #

Cвязь с другими мобильными сетями, 3, 0.50, мин

Роуминг, 4, 10, мин

Международный тариф, 5, 50, мин

Файл информации об услугах, оказанных клиентам, включает в себя записи, состоящие из полей номера телефона, кода услуги, даты и времени ее использования в секундах. Знак # используется в случае, если время не определяется (например, отправляется СМС). Пример содержимого в файле:

9734672311, 1, 13.02.2008 13:01:55, 300

9734672311, 2, 28.11.2012 01:32:30, #

9757282392, 5, 23.09.2013 19:14:00, 54

**Требования к представлению результатов работы программы**

Каждый файл хранится в том же каталоге, что и разработанная программа.

Формат файлов текстовый, каждый из них создан и редактируется текстовым редактором.

Содержание файла настройки определяется вариантом в таблице индивидуальных заданий (см. ниже).

Параметры настройки записываются в текстовый файл с именем Param.ini. Каждый параметр – в отдельной строке.

Результирующий отчет (результат обработки) записывается в файл с именем Отчёт.txt. Если информация отсутствует, вывести в файл строку «Нет данных».

Вывести сообщение в окно программы о результатах её выполнения – есть или отсутствуют данные в результирующем файле. Язык сообщения английский либо русский на транслите.

Формат данных в файле Отчёт.txt: каждая строка списка отделяется символами перевода строки. Если в строке несколько полей, они разделяются символами « , »

# C:\Users\Vladimir\Downloads\photo1703709260.jpegC:\Users\Vladimir\Downloads\photo1703711091.jpegC:\Users\Vladimir\Downloads\photo1703709244.jpegВыполнение работы (БЛОК-СХЕМА)

Рисунок 1 – Блок-схема программы ч.

Рисунок 2 – Блок-схема программы ч.2

Рисунок 3 – Блок-схема программы ч.3

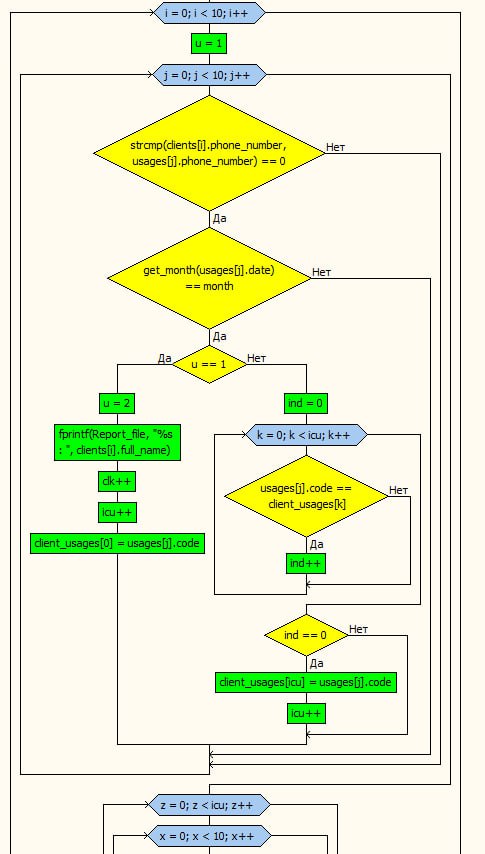
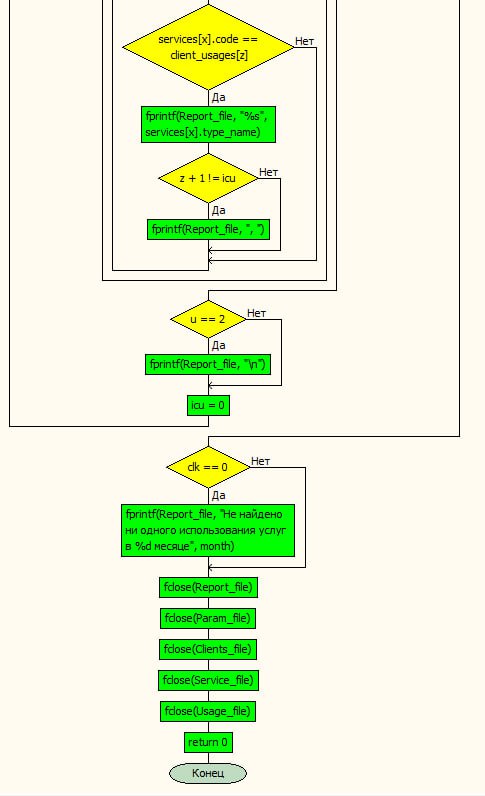
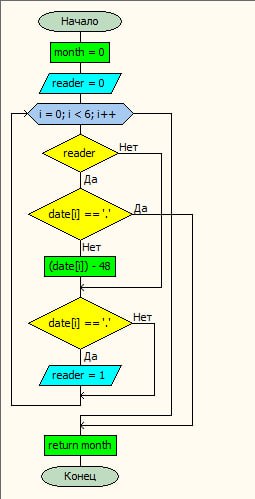


Рисунок 6 – Блок-схема подпрограммы

Рисунок 5 – Блок-схема программы ч.5

Рисунок 4 – Блок-схема программы ч.4

# Исходный код программы

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <locale.h>

struct Client {

char full\_name[200]; // фамилию, имя, отчество клиента

char phone\_number[100]; // номер телефона

char start\_date[10]; // дата заключения договора

char end\_date[10]; // дата окончания договора

float debt; // размер задолженности

float credit; // допустимый кредит

};

struct Service {

char type\_name[100]; // название услуги

int code; // код услуги

float tariff; // тарифф

char unit[10]; // единица измерения

};

struct Usage {

char phone\_number[100]; // номер телефона

int code; // код услуги

char date[100]; // дата

char time[100]; // время

char volume[100]; // обьем использования

};

int get\_month(char\* date) {

int month = 0;

int reader = 0;

for (int i = 0; i < 6; i++) {

if (reader) {

if (date[i] == '.') {

break;

}

month = month \* 10 + static\_cast<int>(date[i]) - 48;

}

if (date[i] == '.') {

reader = 1;

}

}

return month;

}

int main() {

setlocale(LC\_ALL, "Rus");

struct Client clients[1000];

struct Service services[1000];

struct Usage usages[1000];

FILE\* Param\_file = fopen("Param.ini", "r");

if (Param\_file == NULL) {

printf("Не удается открыть файл Param.ini");

}

FILE\* Report\_file = fopen("Report.txt", "w");

FILE\* Clients\_file = fopen("Clients.txt", "r");

if (Clients\_file == NULL) {

printf("Не удается открыть файл Clients.txt");

}

FILE\* Service\_file = fopen("Service.txt", "r");

if (Service\_file == NULL) {

printf("Не удается открыть файл Service.txt");

}

FILE\* Usage\_file = fopen("Usage.txt", "r");

if (Usage\_file == NULL) {

printf("Не удается открыть файл Usage.txt");

}

char smonth[10];

fgets(smonth, 2, Param\_file);

int month = atoi(smonth);

int i = 0;

char str[1000];

while (fgets(str, 1000, Clients\_file)) { // Чтение из файла Clients.txt

char\* buffer = strtok(str, ",");

strcpy(clients[i].full\_name, buffer);

buffer = strtok(NULL, ",");

strcpy(clients[i].phone\_number, buffer);

buffer = strtok(NULL, ",");

strcpy(clients[i].start\_date, buffer);

buffer = strtok(NULL, ",");

strcpy(clients[i].end\_date, buffer);

buffer = strtok(NULL, ",");

clients[i].debt = atof(buffer);

buffer = strtok(NULL, ",");

clients[i].credit = atof(buffer);

i++;

}

i = 0;

while (fgets(str, 1000, Service\_file)) { // Чтение из файла Service.txt

char\* buffer = strtok(str, ",");

strcpy(services[i].type\_name, buffer);

buffer = strtok(NULL, ",");

services[i].code = atoi(buffer);

buffer = strtok(NULL, ",");

services[i].tariff = atof(buffer);

buffer = strtok(NULL, ",");

strcpy(services[i].unit, buffer);

i++;

}

i = 0;

while (fgets(str, 1000, Usage\_file)) { // Чтение из файла Service.txt

char\* buffer = strtok(str, ",");

strcpy(usages[i].phone\_number, buffer);

buffer = strtok(NULL, ",");

usages[i].code = atoi(buffer);

buffer = strtok(NULL, " ");

strcpy(usages[i].date, buffer);

buffer = strtok(NULL, ",");

strcpy(usages[i].time, buffer);

buffer = strtok(NULL, ",");

strcpy(usages[i].volume, buffer);

i++;

}

int client\_usages[10]; // коды услуг которые использовал конкретный пользователь

int icu = 0; // итератор массива кодов

int ind = 0; // совпадения услуг

int u = 1; // проверка на вывод ФИО и \n после

int clk = 0; // количество найденных клиентов

for (int i = 0; i < 10; i++) {

u = 1;

for (int j = 0; j < 10; j++) {

if (strcmp(clients[i].phone\_number, usages[j].phone\_number) == 0) {

if (get\_month(usages[j].date) == month) {

if (u == 1) {

u = 2;

fprintf(Report\_file, "%s : ", clients[i].full\_name);

clk++;

icu++;

client\_usages[0] = usages[j].code;

}

else {

ind = 0;

for (int k = 0; k < icu; k++) {

if (usages[j].code == client\_usages[k]) {

ind++;

}

}

if (ind == 0) {

client\_usages[icu] = usages[j].code;

icu++;

}

}

}

}

}

for (int z = 0; z < icu; z++) {

for (int x = 0; x < 10; x++) {

if (services[x].code == client\_usages[z]) {

fprintf(Report\_file, "%s", services[x].type\_name);

if (z + 1 != icu) { // постановка запятой кроме последней

fprintf(Report\_file, ", ");

}

}

}

}

if (u == 2) {

fprintf(Report\_file, "\n");

}

icu = 0;

}

if (clk == 0) {

fprintf(Report\_file, "Не найдено ни одного использования услуг в %d месяце", month);

}

fclose(Report\_file);

fclose(Param\_file);

fclose(Clients\_file);

fclose(Service\_file);

fclose(Usage\_file);

return 0;

}

# Пример работы программы

На рисунках 11-15 представлены данный входных файлов.

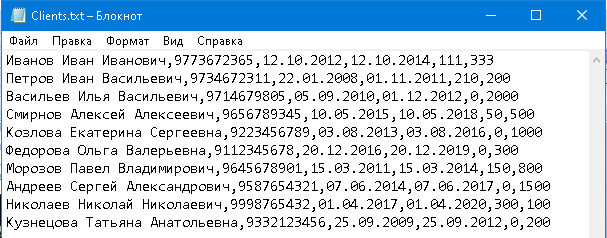


Рисунок 11 – Файл Clients.txt

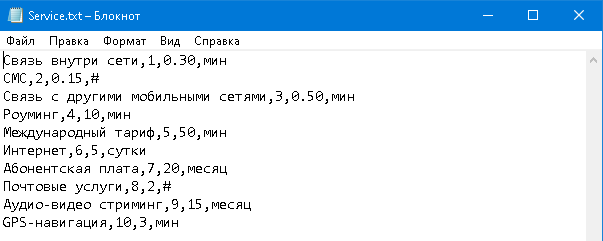


Рисунок 12 – Файл Service.txt

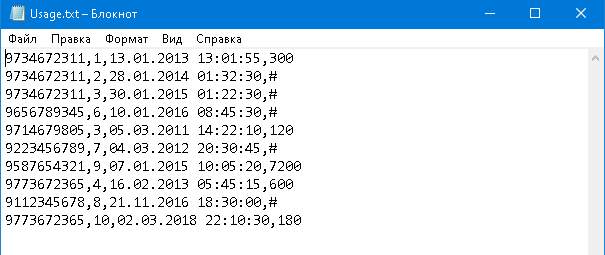


Рисунок 13 – Usage.txt



Рисунок 14 – Param.ini

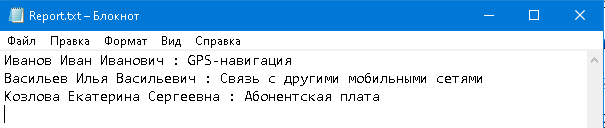


Рисунок 15 – Report.txt

Из рисунков мы можем убедиться в том, что программа работает корректно, исходя из данных в файлах видно, что пользователи Иванов Иван Иванович использовал данные услуги: GPS-навигация в указанном месяце.

Васильев Илья Васильевич: Связь с другими мобильными сетями. Из чего следует, что в файл Report.txt записываются все клиенты, которые пользовались услугами в указанный месяц и все услуги, которыми пользовались эти клиенты в указанный месяц.

# Вывод

В ходе выполнения индивидуального, индивидуального задания, я разработал программу, которая, обрабатывая три текстовых файла, записывает всех клиентов, пользовавшихся услугами в определенный месяц, приводя список услуг которыми они пользовались.

При выполнении индивидуального задания я научился закрепил знания и навыки такие как: работа с файлами, работа со структурами, циклы, обработка строк, работа с массивами, работа с использованием русскоязычной кодировки.