Министерство науки и высшего образования Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ

ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет Безопасности Информационных Технологий

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

«Управление мобильными устройствами»

Выполнил Студент,

Группы N3350

Находкин Александр Михайлович

Подпись:

Проверил: доцент ФБИТ,

Университет ИТМО,

Федоров Иван Романович

Подпись:

Санкт-Петербург

2020

# Цель работы:

Изучение биллинговой системы и тарификации звонков.

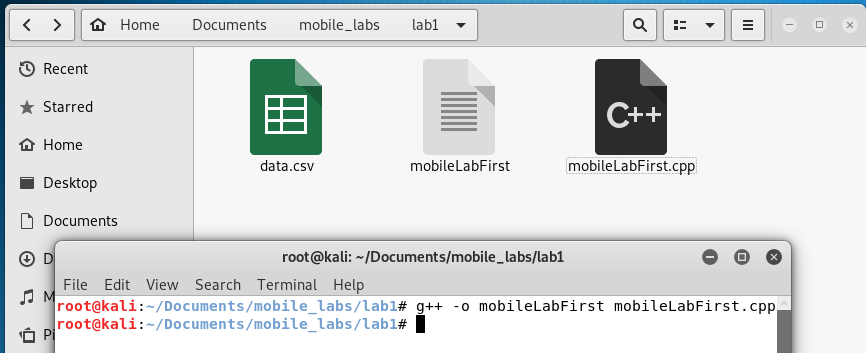
# Задачи:

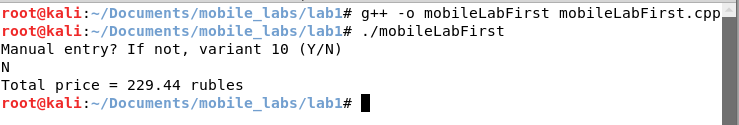
Реализация простейшего правила тарификации для услуг типа “Телефония” по длительности разговора и “СМС” по общему количеству.

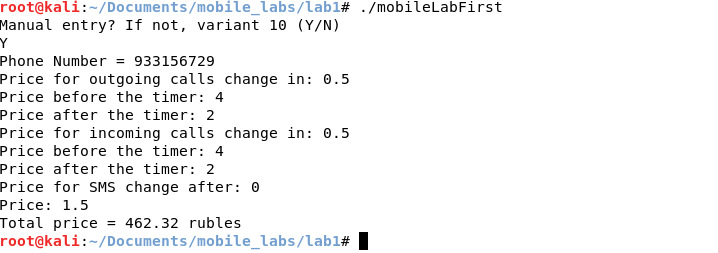
# Реализация

Программа была реализована на языке C++ из-за достаточных знаний этого языка для выполнения данной работы.

# Выполнение программы







# Выводы

Во время выполнения данной лабораторной работы, были реализованы простейшие правила тарификации.

# Исходный код

Исходный код можно найти на <https://github.com/larentoun/ITMO_Mobiles_N3350_NakhodkinAM>

Исходный код:

#include <iostream>

#include <string>

#include <sstream>

#include <fstream>

#include <vector>

using namespace std;

float read\_data(string phoneNumber, int callReceivePriceA, int callReceivePriceB, float callReceivePriceTimer, int callPriceA, int callPriceB, float callPriceTimer, int smsPriceA, int smsPriceB, int smsPriceCounter)

{

float totalPrice = 0;

//open file

fstream fileToRead;

fileToRead.open("data.csv", ios::in);

//create a vector which will be written

vector<string> row(5);

string line, word, temp;

//reading every row and comparing it to the request

while (fileToRead >> temp)

{

row.clear();

getline(fileToRead, line);

stringstream s(line);

// read every column data of a row and

// store it in a string variable, 'word'

while (getline(s, word, ','))

{

row.push\_back(word);

}

//row[0] = timestamp

//row[1] = caller

//row[2] = receiver

//row[3] = call duration

//row[4] = nummber of SMS

//Change totalPrice if needed

if (row[1] == phoneNumber)

{

if (stof(row[3]) > callPriceTimer)

{

totalPrice = totalPrice + callPriceTimer \* callPriceA;

totalPrice = totalPrice + (stof(row[3]) - callPriceTimer) \* callPriceB;

}

else

{

totalPrice = totalPrice + stof(row[3]) \* callPriceB;

}

if (stoi(row[4]) > smsPriceCounter)

{

totalPrice = totalPrice + smsPriceCounter \* smsPriceA;

totalPrice = totalPrice + (stof(row[4]) - smsPriceCounter) \* smsPriceB;

}

else

{

totalPrice = totalPrice + stof(row[4]) \* smsPriceB;

}

}

if (row[2] == phoneNumber)

{

if (stof(row[3]) > callPriceTimer)

{

totalPrice = totalPrice + callPriceTimer \* callReceivePriceA;

totalPrice = totalPrice + (stof(row[3]) - callPriceTimer) \* callReceivePriceB;

}

else

{

totalPrice = totalPrice + stof(row[3]) \* callReceivePriceB;

}

}

}

fileToRead.close();

return totalPrice;

}

int main()

{

float totalPrice = 0;

string phoneNumber;

int callReceivePriceA = 0;

int callReceivePriceB = 0;

float callReceivePriceTimer = 0;

int callPriceA = 0;

int callPriceB = 0;

float callPriceTimer = 0;

int smsPriceA = 0;

int smsPriceB = 0;

int smsPriceCounter = 0;

phoneNumber = "933156729";

callPriceTimer = 0;

callPriceA = 4;

callPriceB = 2;

callReceivePriceTimer = 0;

callReceivePriceA = 0;

callReceivePriceB = 0;

smsPriceCounter = 10;

smsPriceA = 0;

smsPriceB = 1;

string ifManual;

cout << "Manual entry? If not, variant 10 (Y/N)" << endl;

cin >> ifManual;

if (ifManual == "N")

{

totalPrice = read\_data(phoneNumber, callReceivePriceA, callReceivePriceB, callReceivePriceTimer, callPriceA, callPriceB, callPriceTimer, smsPriceA, smsPriceB, smsPriceCounter);

cout << "Total price = " << totalPrice << " rubles";

return 0;

}

//Input Phone Number

cout << "Phone Number = ";

cin >> phoneNumber;

//Input Outgoing Call Prices

cout << "Price for outgoing calls change in: ";

cin >> callPriceTimer;

if (callPriceTimer == 0)

{

cout << "Price: ";

cin >> callPriceB;

}

else

{

cout << "Price before the timer: ";

cin >> callPriceA;

cout << "Price after the timer: ";

cin >> callPriceB;

}

//Input Receive Call Prices

cout << "Price for incoming calls change in: ";

cin >> callReceivePriceTimer;

if (callReceivePriceTimer == 0)

{

cout << "Price: ";

cin >> callReceivePriceB;

}

else

{

cout << "Price before the timer: ";

cin >> callReceivePriceA;

cout << "Price after the timer: ";

cin >> callReceivePriceB;

}

//Input SMS Prices

cout << "Price for SMS change after: ";

cin >> smsPriceCounter;

if (smsPriceCounter == 0)

{

cout << "Price: ";

cin >> smsPriceB;

}

else

{

cout << "Price before the counter: ";

cin >> smsPriceA;

cout << "Price after the counter: ";

cin >> smsPriceB;

}

totalPrice = read\_data(phoneNumber, callReceivePriceA, callReceivePriceB, callReceivePriceTimer, callPriceA, callPriceB, callPriceTimer, smsPriceA, smsPriceB, smsPriceCounter);

cout << "Total price = " << totalPrice << " rubles" << endl;

return 0;

}