**Министерство науки и высшего образования РФ**

Федеральное государственное автономное

образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

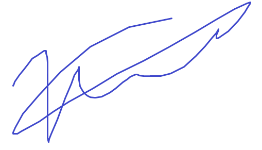
факультет БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ

ТЕХНОЛОГИЙ

Управление мобильными устройствами

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3**

«Формирование счета на оплату услуг»

Выполнил: студент

группы N3349

Царев Иван Павлович

Проверил: инженер ФБИТ,

Университет ИТМО,

Федоров Иван Романович

Санкт-Петербург

2020

**Цель:** cформировать счет на оплату в формате .pdf по полученным результатам тарификации услуг «Телефония» и «Интернет» в лабораторных работах 1, 2.

**Задачи:**

* сформировать шаблон счета на оплату;
* вывести результат лабораторных работ 1, 2 в файлы в папку с данной лабораторной работой;
* сформировать счет.

**Описание выбранных средств реализации:** для формирования счета на оплату был выбран язык программирования Си. ОС: Ubuntu 20.04 (64 bit), утилита unoconv. Вероятно, выбранный язык является не самым удобным для данной работы, но этот же язык был использован в предыдущих работах. Это дает возможность создать единую программу для обработки, тарификации услуг и формирования счета на оплату.

**Описание работы программы:**

Изначально имеется папка с 5 файлами: script, lr3.c, temp.txt (шаблон), lr1.txt (результат л/р №1), lr2.txt (результат л/р №2).

Запускается с помощью ./script. Данный скрипт создает файл num.txt, в котором будет храниться номер последнего счета на оплату, затем компилирует и запускает программу lr3, которая обрабатывает полученные ранее результаты лабораторных работ 1, 2 Потом скрипт преобразует созданный текстовый документ в формат .pdf с помощью утилиты unoconv. После всей работы скрипт удаляет все созданные файлы, кроме счета на оплату в формате .pdf и файла num.txt.

Текст скрипта script и программы lr3.c находятся в приложении к данной лабораторной работе.

**Выводы:** на практике с помощью языка Си и утилиты unoconv был сформирован счет на оплату в формате .pdf по полученным результатам тарификации услуг «Телефония» и «Интернет» в предыдущих лабораторных работах.

**Приложение**

Файл script:

#!/bin/bash  
  
» num.txt  
gcc lr3.c -o lr3  
./lr3  
unoconv -f pdf bill.txt  
rm lr3 bill.txt

Файл lr3.c:

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <time.h>

void number\_in\_words(double, char\*); //функция для получения суммы прописью

int main (int argc,char \*argv[]) {

FILE \*new, \*txt, \*num, \*lr1, \*lr2;

int j = 0, count;

char sym, temp[100];

char inn[13] = "123456789012", kpp[10] = "123456789", date[11], number[5], unit1[4] = "min", price1[8] = "1", unit2[4] = "pcs", price2[8] = "0/1/2", unit3[3] = "Kb", price3[8] = "0.5/1", total\_in\_words[200], client[100] = "Клиентов Клиент Клиентович";

new = fopen("bill.txt", "w"); //счет на оплату

txt = fopen("temp.txt", "r"); //шаблон счета

lr1 = fopen("lr1.txt", "r"); //телефония и смс (1 л/р)

lr2 = fopen("lr2.txt", "r"); //трафик (2 л/р)

num = fopen("num.txt", "r"); //счетчик документов

while ((sym=fgetc(txt)) != EOF) {

fputc(sym, new);

}

while ((sym=fgetc(num)) != EOF) {

number[j] = sym;

j++;

}

count = atoi(number);

count++;

freopen("num.txt", "w", num); //+1 в счетчик перезаписать

fprintf(num,"%d", count);

long int s\_time = time(NULL); //для даты

struct tm \*m\_time = localtime(&s\_time);

strftime(date,11,"%d.%m.%Y",m\_time);

fgets(temp,100,lr1);

double dur = atof(temp); //продолжительность звонков

fgets(temp,100,lr1);

double amount1 = atof(temp); //сумма за звонки

fgets(temp,100,lr1);

int sms = atoi(temp); //кол-во смс

fgets(temp,100,lr1);

double amount2 = atof(temp); //сумма за смс

fgets(temp,100,lr2);

double traffic = atof(temp); //кол-во интернета

fgets(temp,100,lr2);

double amount3 = atof(temp); //сумма за интернет

double amountall = amount1 + amount2 + amount3; //всего

double nds = amountall \* 0.2;

double total = amountall + nds; //итого

number\_in\_words(total, total\_in\_words);

fseek(new, 893, SEEK\_SET); //ИНН

fprintf(new,"%s",inn);

fseek(new, 922, SEEK\_SET); //КПП

fprintf(new,"%s",kpp);

fseek(new, 1753, SEEK\_SET); //Номер счета на оплату

fprintf(new,"%d", count);

fseek(new, 1764, SEEK\_SET); //дата

fprintf(new,"%s",date);

fseek(new, 2188, SEEK\_SET); //Заказчик

fprintf(new,"%s",client);

fseek(new, 3037, SEEK\_SET); //Кол-во1

fprintf(new,"%.2lf", dur);

fseek(new, 3048, SEEK\_SET); //Ед1

fprintf(new,"%s",unit1);

fseek(new, 3059, SEEK\_SET); //Цена1

fprintf(new,"%s",price1);

fseek(new, 3073, SEEK\_SET); //Сумма1

fprintf(new,"%.2lf",amount1);

fseek(new, 3376, SEEK\_SET); //Кол-во2

fprintf(new,"%d",sms);

fseek(new, 3387, SEEK\_SET); //Ед2

fprintf(new,"%s",unit2);

fseek(new, 3397, SEEK\_SET); //Цена2

fprintf(new,"%s",price2);

fseek(new, 3412, SEEK\_SET); //Сумма2

fprintf(new,"%.2lf",amount2);

fseek(new, 3721, SEEK\_SET); //Кол-во3

fprintf(new,"%.1lf",traffic);

fseek(new, 3732, SEEK\_SET); //Ед3

fprintf(new,"%s",unit3);

fseek(new, 3742, SEEK\_SET); //Цена3

fprintf(new,"%s",price3);

fseek(new, 3757, SEEK\_SET); //Сумма3

fprintf(new,"%.2lf",amount3);

fseek(new, 4078, SEEK\_SET); //Итого

fprintf(new,"%.2lf руб",amountall);

fseek(new, 4171, SEEK\_SET); //НДС

fprintf(new,"%.2lf руб",nds);

fseek(new, 4264, SEEK\_SET); //Всего

fprintf(new,"%.2lf руб",total);

fseek(new, 4362, SEEK\_SET); //Прописью

fprintf(new,"%s", total\_in\_words);

fclose(new);

fclose(num);

fclose(txt);

return 0;

}

void number\_in\_words(double num, char \*s)

{

int position = 0; // позиция в строке

int x[6]; // массив разрядов (хххх руб хх коп)

int div = 100000; // делитель на разряды

int var = num \* 100; // частное от деления

int rest; // остаток от деления

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

{

rest = (int)(var / div);

x[i] = rest;

var = var - rest\*div;

div = div / 10;

}

// Тысячи

switch (x[0])

{

case 1: strcpy(s, "Одна тысяча "); break;

case 2: strcpy(s, "Две тысячи "); break;

case 3: strcpy(s, "Три тысячи "); break;

case 4: strcpy(s, "Четыре тысячи "); break;

case 5: strcpy(s, "Пять тысяч "); break;

case 6: strcpy(s, "Шесть тысяч "); break;

case 7: strcpy(s, "Семь тысяч "); break;

case 8: strcpy(s, "Восемь тысяч "); break;

case 9: strcpy(s, "Девять тысяч "); break;

default: break;

}

position = strlen(s);

// Сотни

switch (x[1])

{

case 1: strcpy(s + position, "сто "); break;

case 2: strcpy(s + position, "двести "); break;

case 3: strcpy(s + position, "триста "); break;

case 4: strcpy(s + position, "четыреста "); break;

case 5: strcpy(s + position, "пятьсот "); break;

case 6: strcpy(s + position, "шестьсот "); break;

case 7: strcpy(s + position, "семьсот "); break;

case 8: strcpy(s + position, "восемьсот "); break;

case 9: strcpy(s + position, "девятьсот "); break;

default: break;

}

position = strlen(s);

// Десятки

switch (x[2])

{

case 1:

// Числа от 10 до 19

switch (x[3])

{

case 0: strcpy(s + position, "десять "); break;

case 1: strcpy(s + position, "одиннадцать "); break;

case 2: strcpy(s + position, "двенадцать "); break;

case 3: strcpy(s + position, "тринадцать "); break;

case 4: strcpy(s + position, "четырнадцать "); break;

case 5: strcpy(s + position, "пятнадцать "); break;

case 6: strcpy(s + position, "шестнадцать "); break;

case 7: strcpy(s + position, "семнадцать "); break;

case 8: strcpy(s + position, "восемнадцать "); break;

case 9: strcpy(s + position, "девятнадцать "); break;

}

x[3] = 0;

position = strlen(s);

break;

case 2: strcpy(s + position, "двадцать "); break;

case 3: strcpy(s + position, "тридцать "); break;

case 4: strcpy(s + position, "сорок "); break;

case 5: strcpy(s + position, "пятьдесят "); break;

case 6: strcpy(s + position, "шестьдесят "); break;

case 7: strcpy(s + position, "семьдесят "); break;

case 8: strcpy(s + position, "восемьдесят "); break;

case 9: strcpy(s + position, "девяносто "); break;

default: break;

}

position = strlen(s);

// Единицы

switch (x[3]) {

case 1: strcpy(s + position, "один "); break;

case 2: strcpy(s + position, "два "); break;

case 3: strcpy(s + position, "три "); break;

case 4: strcpy(s + position, "четыре "); break;

case 5: strcpy(s + position, "пять "); break;

case 6: strcpy(s + position, "шесть "); break;

case 7: strcpy(s + position, "семь "); break;

case 8: strcpy(s + position, "восемь "); break;

case 9: strcpy(s + position, "девять "); break;

default: break;

}

position = strlen(s);

strcpy(s + position, "руб ");

position = strlen(s);

//Копейки

switch (x[4])

{

case 1:

// Числа от 10 до 19

switch (x[5])

{

case 0: strcpy(s + position, "десять "); break;

case 1: strcpy(s + position, "одиннадцать "); break;

case 2: strcpy(s + position, "двенадцать "); break;

case 3: strcpy(s + position, "тринадцать "); break;

case 4: strcpy(s + position, "четырнадцать "); break;

case 5: strcpy(s + position, "пятнадцать "); break;

case 6: strcpy(s + position, "шестнадцать "); break;

case 7: strcpy(s + position, "семнадцать "); break;

case 8: strcpy(s + position, "восемнадцать "); break;

case 9: strcpy(s + position, "девятнадцать "); break;

}

x[5] = 0;

position = strlen(s);

break;

case 2: strcpy(s + position, "двадцать "); break;

case 3: strcpy(s + position, "тридцать "); break;

case 4: strcpy(s + position, "сорок "); break;

case 5: strcpy(s + position, "пятьдесят "); break;

case 6: strcpy(s + position, "шестьдесят "); break;

case 7: strcpy(s + position, "семьдесят "); break;

case 8: strcpy(s + position, "восемьдесят "); break;

case 9: strcpy(s + position, "девяносто "); break;

default: break;

}

position = strlen(s);

// Единицы

switch (x[5]) {

case 1: strcpy(s + position, "один "); break;

case 2: strcpy(s + position, "два "); break;

case 3: strcpy(s + position, "три "); break;

case 4: strcpy(s + position, "четыре "); break;

case 5: strcpy(s + position, "пять "); break;

case 6: strcpy(s + position, "шесть "); break;

case 7: strcpy(s + position, "семь "); break;

case 8: strcpy(s + position, "восемь "); break;

case 9: strcpy(s + position, "девять "); break;

default: break;

}

position = strlen(s);

strcpy(s + position, "коп");

}