Пермский филиал федерального государственного автономного образовательного учреждения высшего образования  
«Национальный исследовательский университет   
Высшая школа экономики»

*Факультет социально-экономических и компьютерных наук*

Мусихин Данил Михайлович

**Работа с памятью**

*Лабораторная работа № 1*

студента образовательной программы «Разработка информационных систем для бизнеса» по направлению подготовки *09.03.04 Программная инженерия*

Руководитель, к.т.н., Доцент кафедры ИТБ.

Л.Н. Лядова

Пермь, 2023 год

**Оглавление**

[1 Постановка задачи 3](#__RefHeading___1)

[2 Анализ задачи 4](#__RefHeading___2)

[2.1 Предметная область 4](#__RefHeading___3)

[2.2 Требования к разрабатываемой системе 4](#__RefHeading___4)

[3 Проектирование 5](#__RefHeading___5)

[3.1 Типы данных 5](#__RefHeading___6)

[3.2 Алгоритмы работы программы 5](#__RefHeading___7)

[4 Код программы 6](#__RefHeading___8)

[5 Тестирование 8](#__RefHeading___9)

[5.1 Тестирование ввода показания 8](#__RefHeading___10)

[5.2 Тестирование подсчета цены за услугу 9](#__RefHeading___11)

# Постановка задачи

Разработайте приложение для начисления оплаты потреблённой электроэнергии. Расчёт выполняется по формуле:

*Начисленная к оплате сумма = (Показания за текущий месяц – Показания за прошлый месяц) × Стоимость киловатт часа*

**Примечание:** Показания вводятся как целые числа, дробная часть не передаётся, хотя может быть показана на приборе учёта. При выполнении задания должны быть выполнены следующие этапы:

1. Анализ требований. Необходимо проанализировать условия решения задачи, включая характеристики приборов учёта, показания которых используются при начислении. Требования фиксируются в техническом задании.
2. Проектирование приложения. На основе требований определяются типы и структуры данных для его реализации, алгоритм, который должен быть реализован.
3. Реализация приложения. Выполняется кодирование, тестирование и отладка программы.

# Анализ задачи

Для анализа опишем предметную область и требования к разрабатываемой программе.

## Предметная область

Для работы использует счетчик электроэнергии. Он считает расход в киловатт/час. Так же у него есть ограничения в счете. Счетчики вмещают от 4 до 8 цифр. Так же производится их замена раз в определенный период.

Еще стоит учитывать разные цены для услуг электроэнергии. В Пермском крае. Цена услуг такова: 3,4 рубля за кВт/ч днем и 2,2 рубля за кВт/ч ночью.

## Требования к разрабатываемой системе

Приложение должно отвечать следующим требованиям:

1. Учет дневного и ночного тарифа.
2. Учет ограничений вмещения цифр в счетчике в выборе типа данных.
3. Учет возможности замены счетчика (то есть сброс счетчика).

# Проектирование

Для устойчивой работы программы нужно учесть 2 пункт: какие типы данных выбираем и алгоритмы для работы программы.

## Типы данных

Первое и единственное, что необходимо учесть, какой тип данных будет использоваться для хранения расхода электроэнергии. Во-первых, это натуральное число. Во-вторых, это число не больше 10^8. Для работы идеально подойдет int.

## Алгоритмы работы программы

Ниже на рисунке 1 приведен алгоритм работы.

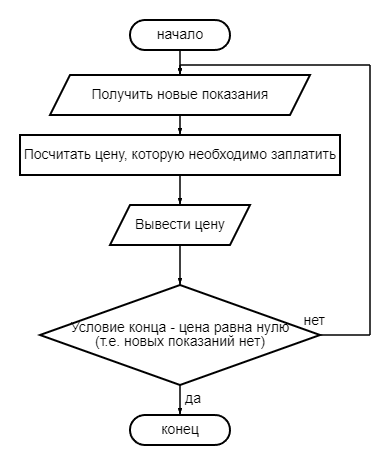


Рисунок 1 - Алгоритм

# Код программы

// ПРОГРАММА ДЛЯ ПОДСЧЕТА КОММУНАЛЬНЫХ УСЛУГ ЗА ЭЛЕКТРИЧЕСТВО

#include <iostream>

#include <iomanip>

#include <string>

#include <limits>

using namespace std;

int \_getValidInput(string sout)

{

const int maxValue = 100000000;

int userInput;

while (true)

{

cout << sout;

if (cin >> userInput)

{

if (userInput < maxValue && userInput >= 0)

return userInput;

else

cout << "Вы ввели нереалистичные показания счетчика. (Число отрицательное, вещественное или >= 100000000)" << endl;

}

else

{

cout << "Вы ввели не число." << endl;

cin.clear(); // очистка флагов ввода

cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // очистка буфера ввода

}

}

}

int main()

{

// цены за кВт/ч

const double priceKWhDay = 3.4;

const double priceKWhNoon = 2.2;

// текущие показания счетчиков

int kWhDay = 0;

int kWhNoon = 0;

// новые показания счетчиков

int newKWhDay = 0;

int newKWhNoon = 0;

// собственно цена за улугу

double price = 0;

// прием новых показаний

cout << "Здравствуйте. Я программа для помощи подсчета комунальных услуг за электричество.\n"

<< "Как только ваши показания перестанут отличаться от предыдуших, тогда я завершу работу." << endl;

do

{

cout << "\nВведите ваш текущие показания за месяц.\n";

newKWhDay = \_getValidInput("День: ");

newKWhNoon = \_getValidInput("Ночь: ");

// подсчет цены

if (newKWhDay >= kWhDay && newKWhNoon >= kWhNoon)

price = (newKWhDay - kWhDay) \* priceKWhDay + (newKWhNoon - kWhNoon) \* priceKWhNoon;

else // счетчики обнулились

{

cout << "Ваш счетчик обнулился вам стоило нам об этом сообщить!" << endl;

price = (newKWhDay - 0) \* priceKWhDay + (newKWhNoon - 0) \* priceKWhNoon;

}

kWhDay = newKWhDay;

kWhNoon = newKWhNoon;

cout << "Ваша цена за текущий месяц = " << fixed << setprecision(2) << price << endl;

} while (price > 0);

}

# Тестирование

Для тестирования используем метод черного ящика.

## Тестирование ввода показания

Ниже на таблице 1 приведены тесты.

Таблица 1 - Тесты

|  |  |  |  |
| --- | --- | --- | --- |
| **Тесты** | **Ввод** | **Ожидаемый результат** | **Полученный результат** |
| **Т1** | 3458 | Присваивание | Присваивание |
| **Т2** | 0 | Присваивание | Присваивание |
| **Т3** | Вылода | Вы ввели не число | Вы ввели не число |
| **Т4** | 0.234 | Вы ввели не натуральное число | Вы ввели не натуральное число |
| **Т5** | 432584325809238590 | Вы ввели слишком большое число | Вы ввели слишком большое число |
| **Т6** | -1 | Вы ввели не натуральное число | Вы ввели не натуральное число |

Проверим полноту приведенных тестов ниже на таблице 2.

Таблица 2 - Проверка полноты тестов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Критерии** | **Т1** | **Т2** | **Т3** | **Т4** | **Т5** | **Т6** |
| **Классы вводимых данных** | | | | | | |
| **Не число** |  |  | + |  |  |  |
| **Вещественное число** |  |  |  | + |  |  |
| **Натуральное число или 0** | + |  |  |  |  |  |
| **Отрицательное** |  |  |  |  |  | + |
| **Натуральное число (>10^8)** |  |  |  |  | + |  |
| **Ноль** |  | + |  |  |  |  |
| **Классы выводимых данных** | | | | | | |
| **Присваивание** | + | + |  |  |  |  |
| **«Вы ввели не число»** |  |  | + |  |  |  |
| **«Вы ввели не натуральное число»** |  |  |  | + |  | + |
| **«Вы ввели слишком большое число»** |  |  |  |  | + |  |

## Тестирование подсчета цены за услугу

Проведем данное тестирование ниже на таблице 3. Будем использовать прошлые показания и новые.

Таблица 3 - Тесты

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Тесты** | **Прошлые показания** | **Новые показания** | **Ожидаемый результат** | **Полученный результат** |
| **Т1** | 0 0 | 10 10 | Ваша цена за текущий месяц = 56.00 | Ваша цена за текущий месяц = 56.00 |
| **Т2** | 453 95 | 32 5 | Ваш счетчик обнулился вам стоило нам об этом сообщить!  Ваша цена за текущий месяц = 119.80 | Ваш счетчик обнулился вам стоило нам об этом сообщить!  Ваша цена за текущий месяц = 119.80 |
| **Т3** | 34 56 | 99999999 99999999 | Ваша цена за текущий месяц = 559999755.60 | Ваша цена за текущий месяц = 559999755.60 |

Проведем проверку полноты тестов ниже на таблице 4.

Таблица 4 - Полнота тестов

|  |  |  |  |
| --- | --- | --- | --- |
| **Критерии** | **Т1** | **Т2** | **Т3** |
| **Классы исключительных ситуаций** | | | |
| **Новый счетчик** |  | + |  |
| **Показания на границе допустимого значения** |  |  | + |