

Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по учебной практике

УП 01.02

Выполнил: Рогов А.М.

Группа: ПР-21

Преподаватель: Мирошниченко Г.В.

2025

Содержание

[1. **Задание №1 Мобильное приложение «Дневник тренировок»** 2](#_Toc118960339)

[1.1 Описание задачи 2](#_Toc118960340)

[1.2 Структура проекта 2](#_Toc118960341)

[1.3 Описание разработанных функций 2](#_Toc118960342)

[1.4 Алгоритм решения 3](#_Toc118960343)

[1.5 Используемые библиотеки 6](#_Toc118960344)

[1.6 Тестовые случаи 6](#_Toc118960345)

[1.7 Используемые инструменты 7](#_Toc118960346)

[1.8 Описание пользовательского интерфейса 7](#_Toc118960347)

[1.9 Приложение (pr screen экранов) 8](#_Toc118960348)

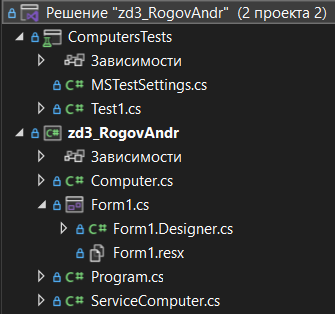
# 1. **Задание №1 Приложение Windows Forms «Компьютеры»**

## Описание задачи

Создать проект для демонстрации работы: ввод-вывод информации об объектах базового класса «Компьютер» и класса потомка «Серверный компьютер». Используя методы Linq и коллекции. Также реализовать модульные тесты к реализованным методам класса.

## Структура проекта

Структура проекта (рис. 1)



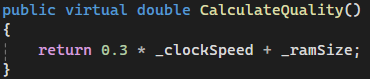
(рис. 1)

В решение входят:

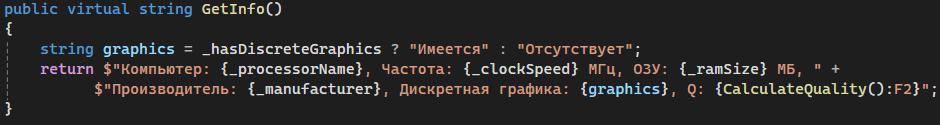
* ComputersTest – модульные тесты к реализованным методам класса.
* zd3\_RogovAndr – проект в котором находятся классы и форма.
* Form1.cs – код, который содержит в себе форма.
* Computer.cs – класс компьютера.
* ServiceComputer.cs – класс серверного компьютера.

## 1.3 Описание разработанных функций

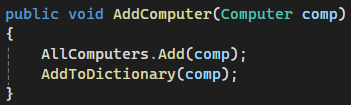
* CalculateQuality() – считает качество компьютера.



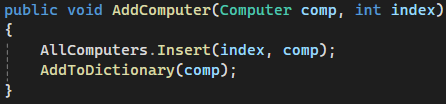
* GetInfo() – выводит информацию о компьютере.



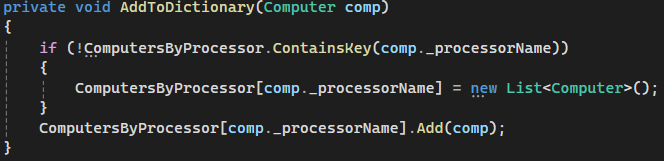
* AddComputer(Computer comp) – добавляет компьютер в список.



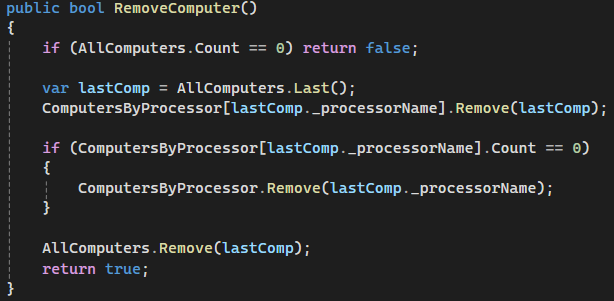
* AddComputer(Computer comp, int index) – добавляет компьютер по индексу.



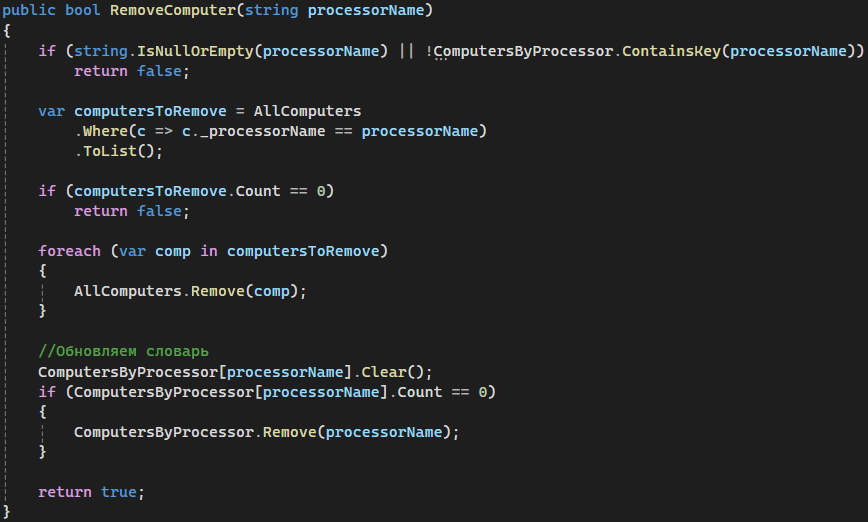
* AddToDictionary(Computer comp) – добавляет в словарь



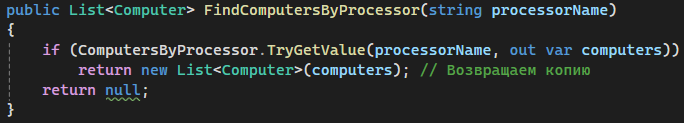
* RemoveComputer() – удаляет последний компьютер.



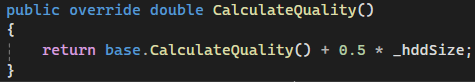
* RemoveComputer(string processorName) – удаляет компьютеры по имени процессора.



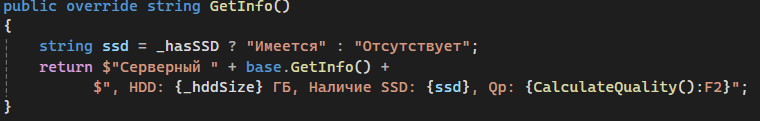
* FindComputersByProcessor() – ищет компьютеры по процессору.



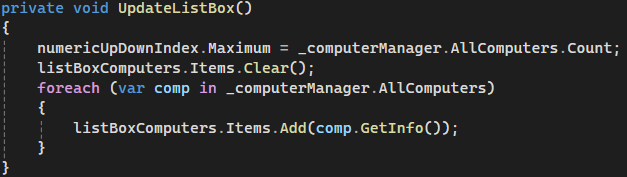
* CalculateQuality() – перегрузка в ServiceComputer, считает Qp с учетом HDD.



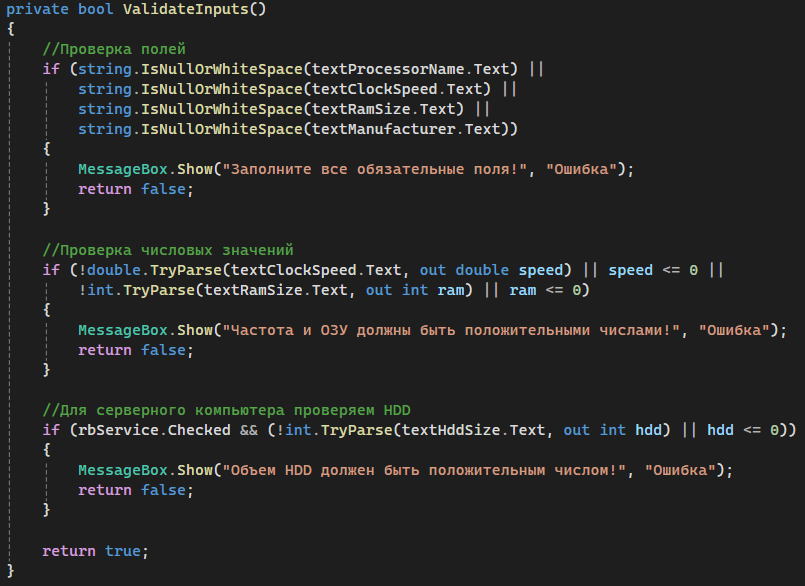
* GetInfo() – перегрузка в ServiceComputer, выводит информацию о серверном компьютере.



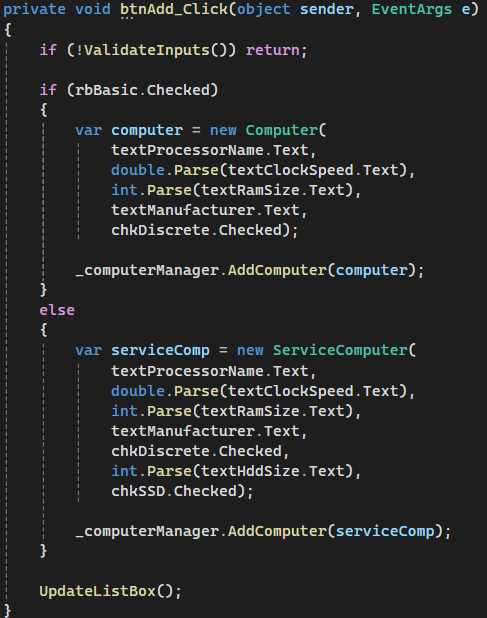
* UpdateListBox() – обновляет листбокс.



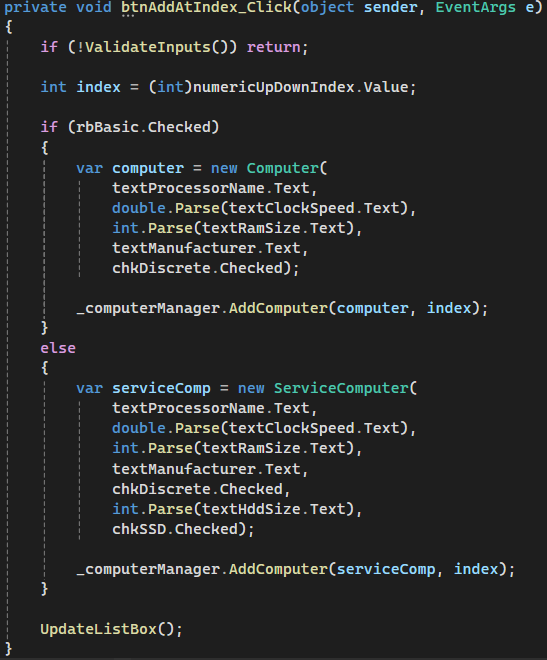
* ValidateInputs() – проверяет, что введены валидные данные.



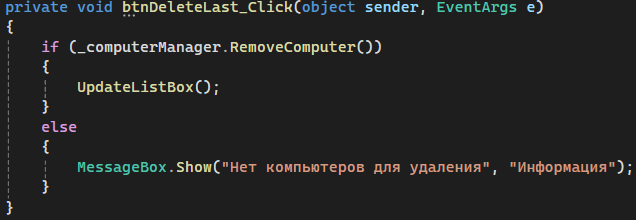
* btnAdd\_Click() – кнопка добавления обычного компа.



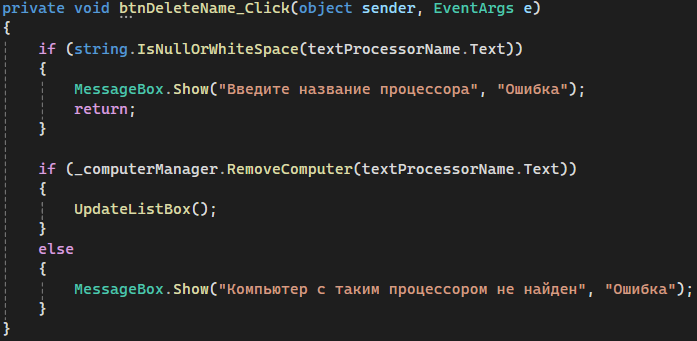
* btnAddAtIndex\_Click() – кнопка добавления по индексу.



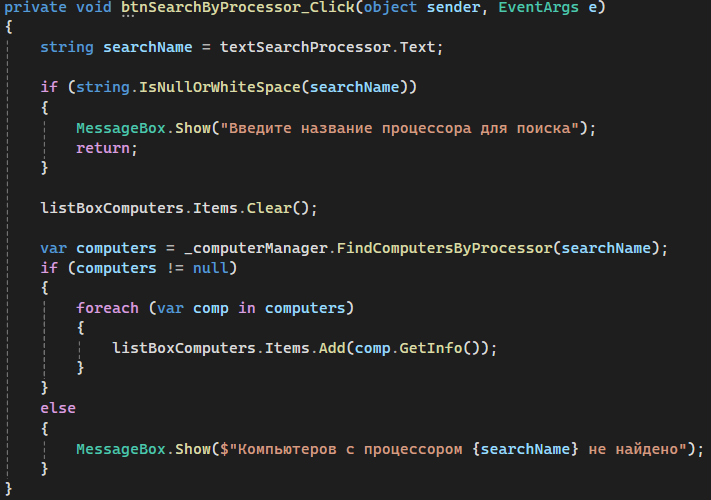
* btnDeleteLast\_Click() – удаляет последний комп.



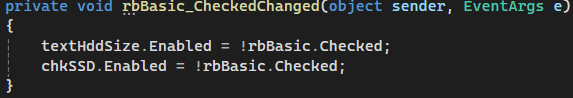
* btnDeleteName\_Click() – удаляет по имени процессора.



* btnSearchByProcessor\_Click() – ищет компьютеры по процессору.



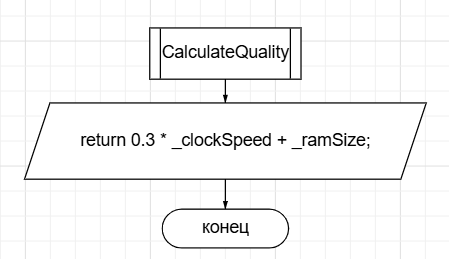
* rbBasic\_CheckedChanged() – меняет доступность полей при переключении радио кнопок.



## Алгоритм решения

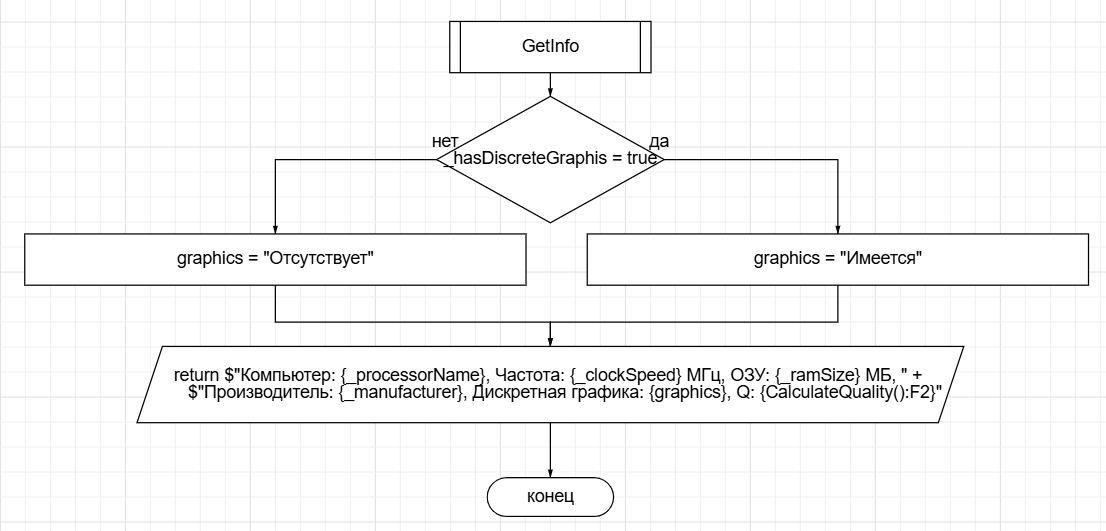
Метод CalculateQuality() (рис. 2)

(рис. 2)



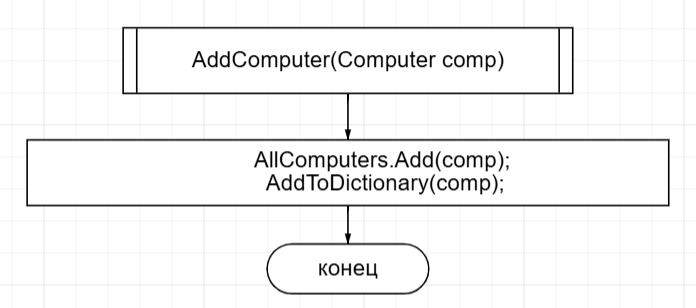
Метод GetInfo() (рис. 3)

(рис. 3)



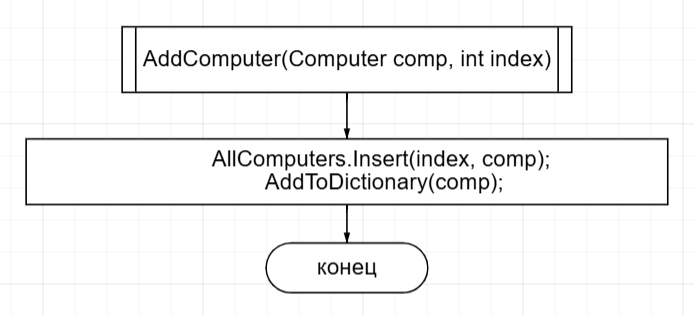
Метод AddComputer(Computer comp) (рис. 4)

(рис. 4)



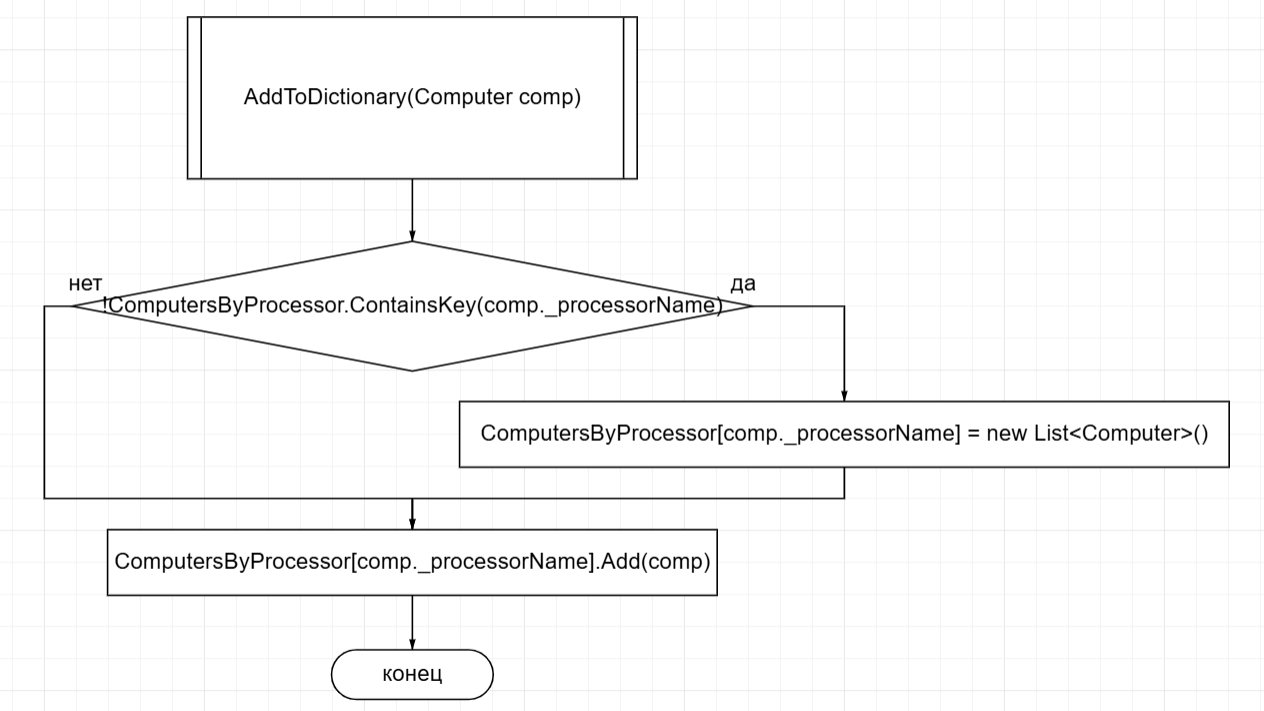
Перегрузка для метода AddComputer(Computer comp, int index) (рис. 5)

(рис. 5)



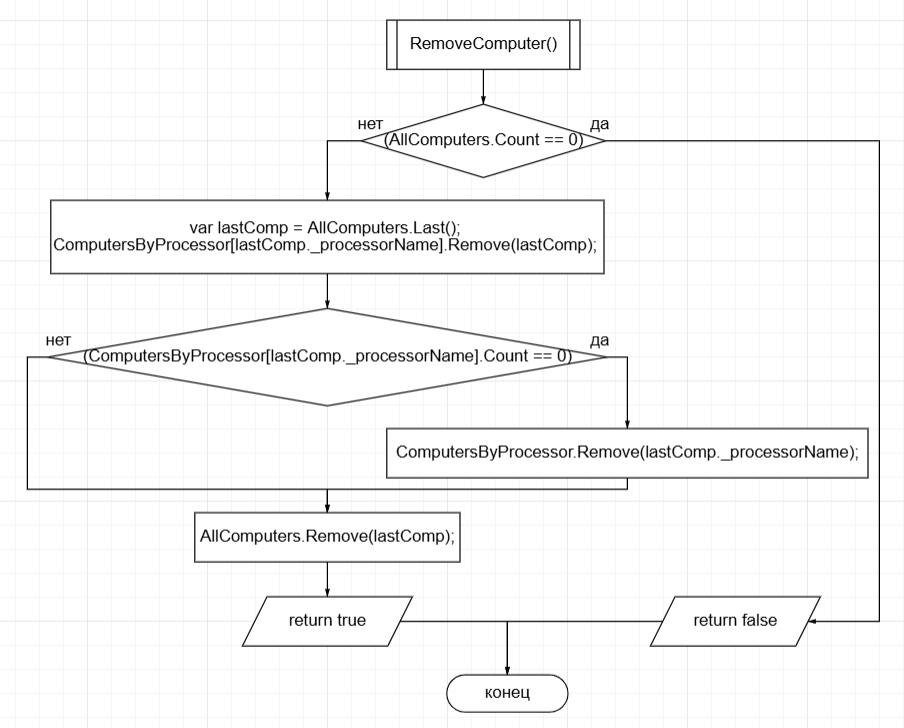
Метод AddToDictionary(Computer comp) (рис. 6)

(рис. 6)



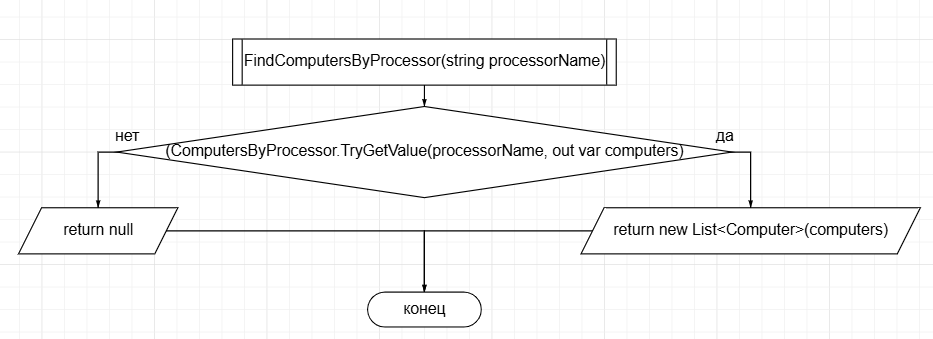
Метод RemoveComputer() (рис. 7)

(рис. 7)



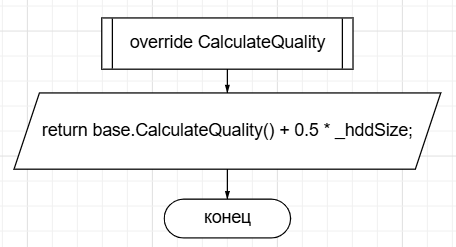
Метод FindComputersByProcessor(string processorName) (рис. 8)

(рис. 8)



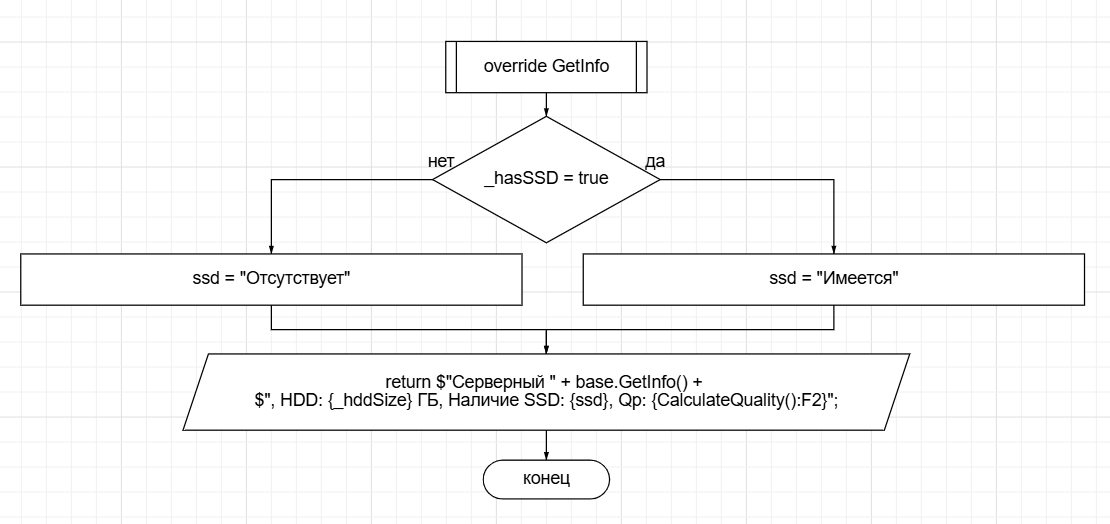
Перегрузка метода CalculateQuality() в классе-потомке (рис. 9)

(рис. 9)



Перегрузка метода GetInfo() в классе-потомке (рис. 10)

(рис. 10)



## Используемые библиотеки

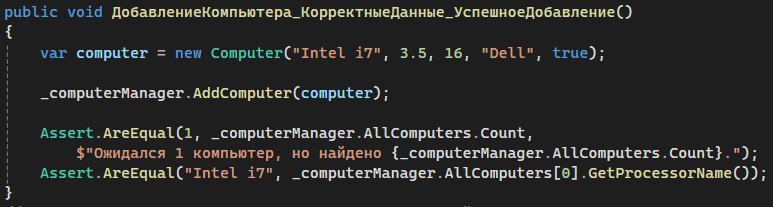
В проекте использовались библиотеки:

* Collections.Generic;
* Linq;
* Text;
* Threading.Tasks;

## 1.6 Тестовые случаи

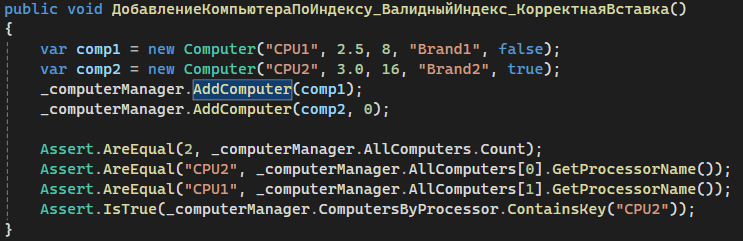
Проверка на добавление компьютера, ожидается появление одного компьютера (рис. 11)

(рис. 11)

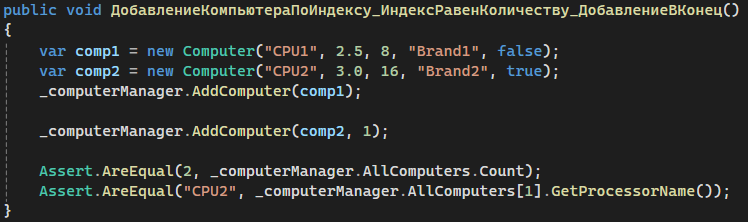


Проверка на добавление компьютера по индексу, ожидается, что добавление по индексу добавит в начало списка (рис. 12)

(рис. 12)



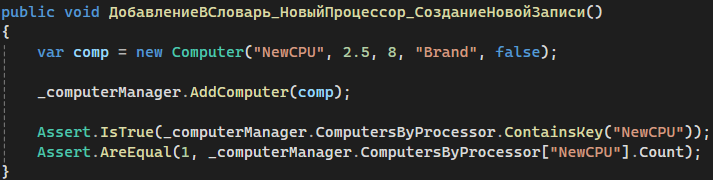
Проверка на добавление по индексу, ожидается добавление в конец списка (рис. 13)



(рис. 13)

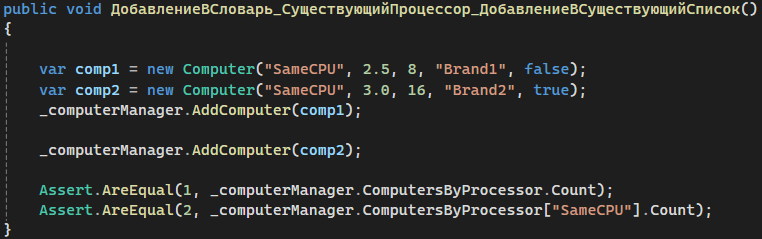
Проверка на добавление в словарь, ожидается добавление в словарь одного компьютера (рис. 14)

(рис. 14)



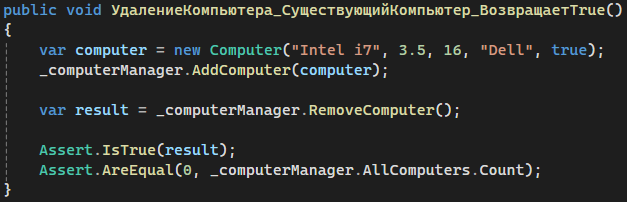
Проверка на добавление в словарь, ожидается увеличение количества объектов до двух (рис. 15)

(рис. 15)



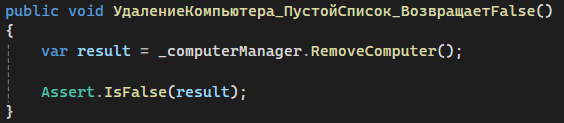
Проверка на удаление компьютера, ожидается True, если список пусто (рис. 16)

(рис. 16)



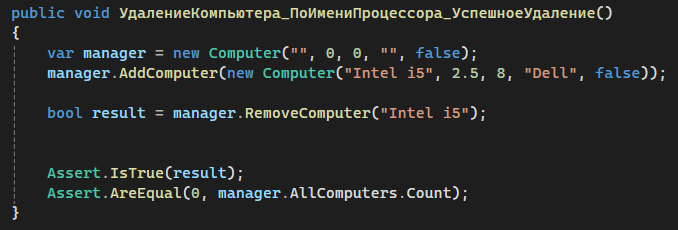
Проверка на удаление компьютера, ожидается false, если список пустой (рис. 17)

(рис. 17)



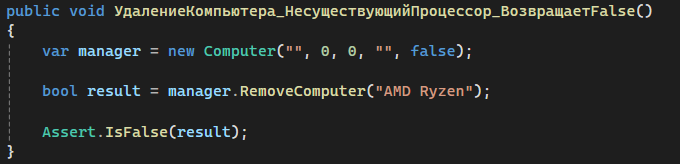
Проверка на удаление компьютера по имени процессора, ожидается true и пустой список (рис. 18)

(рис. 18)



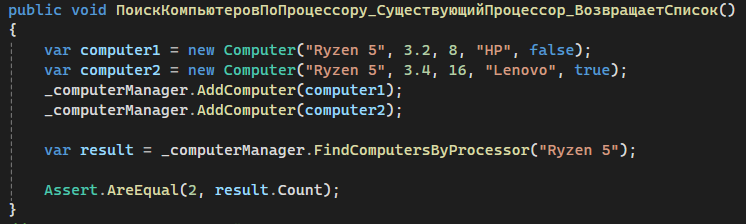
Проверка на удаление компьютера по имени процессора, ожидается false, если не существует компьютера (рис. 19)

(рис. 19)



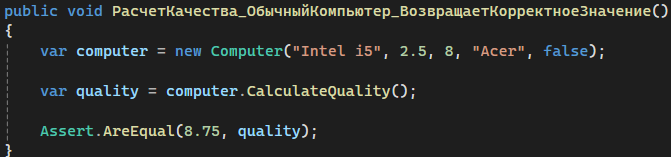
Проверка поиска компьютера по процессору, ожидается список из двух компьютеров с процессором Ryzen 5 (рис. 20)

(рис. 20)



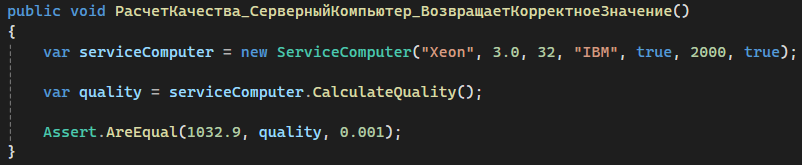
Проверка расчёта качества обычного компьютера, ожидается значение 8.75 (рис. 21)

(рис. 21)



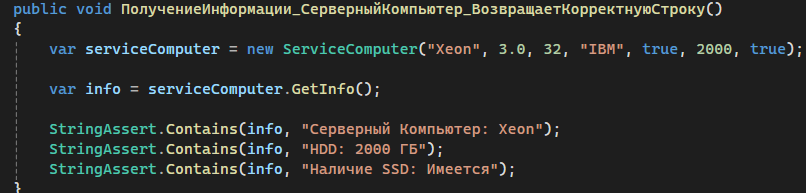
Проверка расчёта качества серверного компьютера, ожидается значение 1032.9 (рис. 22)

(рис. 22)



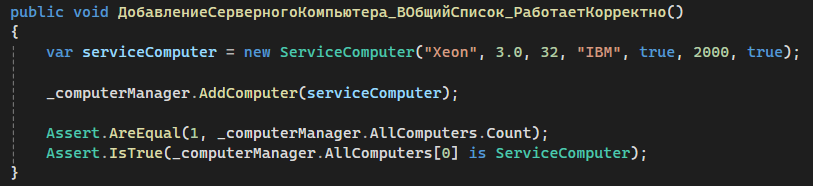
Проверка на получение информации о серверном компьютере, ожидается три определённые строки (рис. 23)

(рис. 23)



Проверка на добавление серверного компьютера в список, ожидается добавление одного компьютера, и его тип равный ServiceComputer (рис. 24)

(рис. 24)

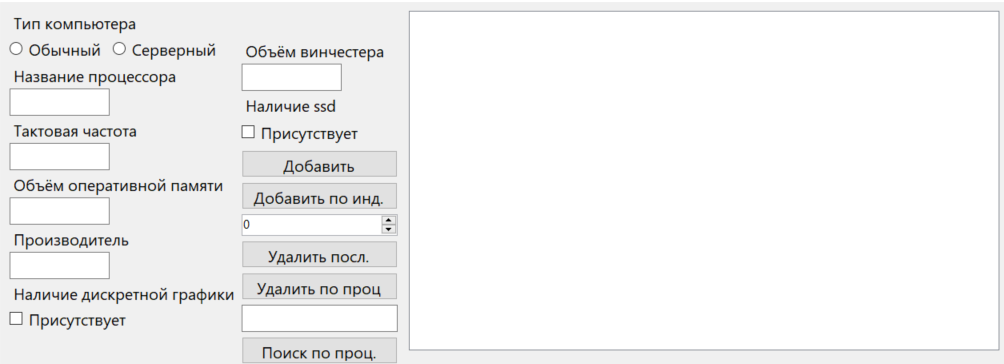


## 1.7 Используемые инструменты

Приложение Windows Forms .Net Framework, C#.

## 1.8 Описание пользовательского интерфейса

Вид формы (рис. 25)



(рис. 25)

Listbox показывает информацию об объектах.

Выбор типа компьютера отвечает за то, какой объект будет создан и передан в listbox.

Кнопка «добавить» добавляет введенные элементы по порядку.

Кнопка «добавить по индексу» смотрит какой выбран индекс в numericUpDown и добавляет по этому индексу в listbox.

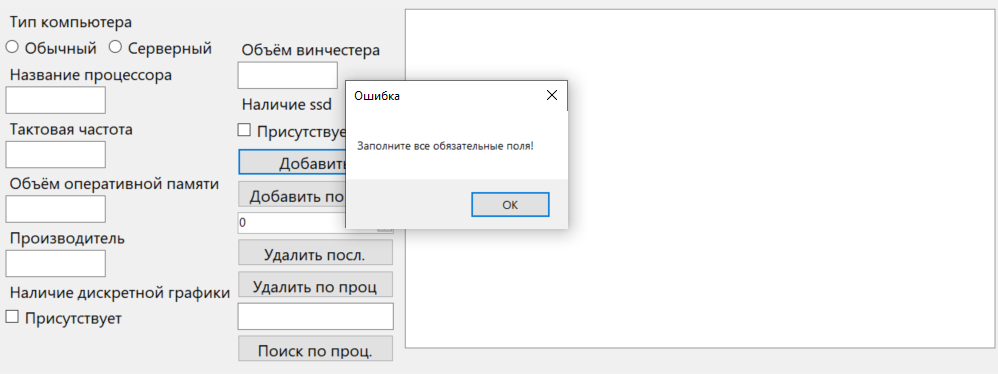
Кнопка «удалить последний» удаляет последний элемент.

Кнопка «удалить по процессору» удаляет компьютеры с выбранным процессором.

Кнопка «поиск по процессору» показывает в listbox компьютеры с подходящим процессором.

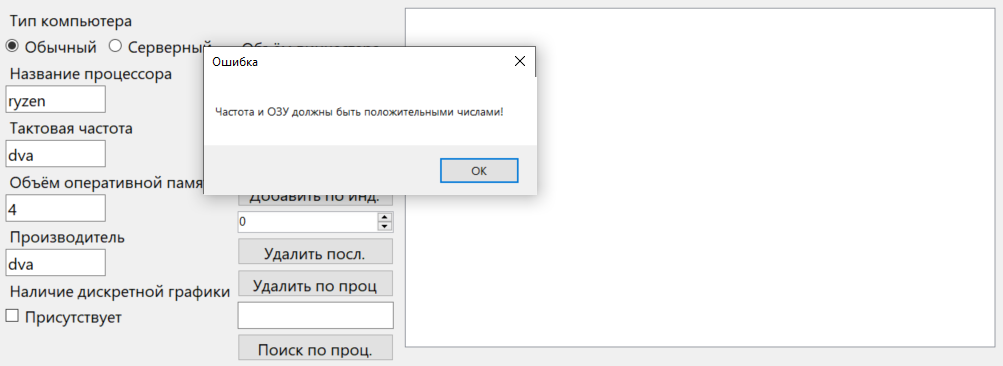
## 1.9 Приложение (pr screen экранов)

Ошибка при пустых полях (рис. 26).



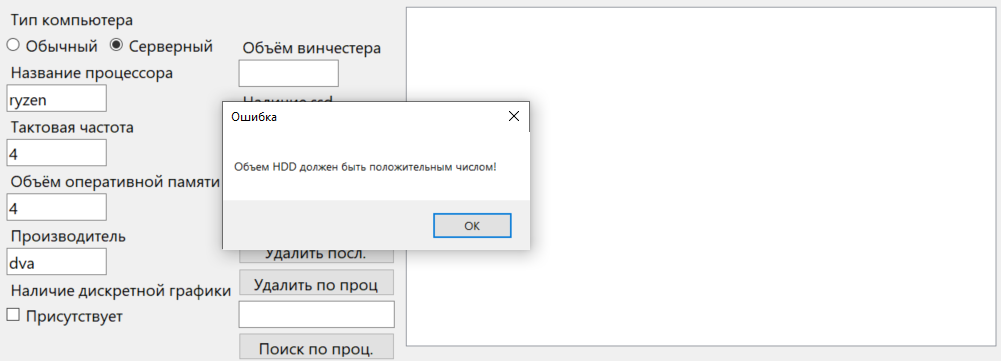
(рис. 26)

Ошибка при не числовом типе в чистоте и ОЗУ (рис. 27).



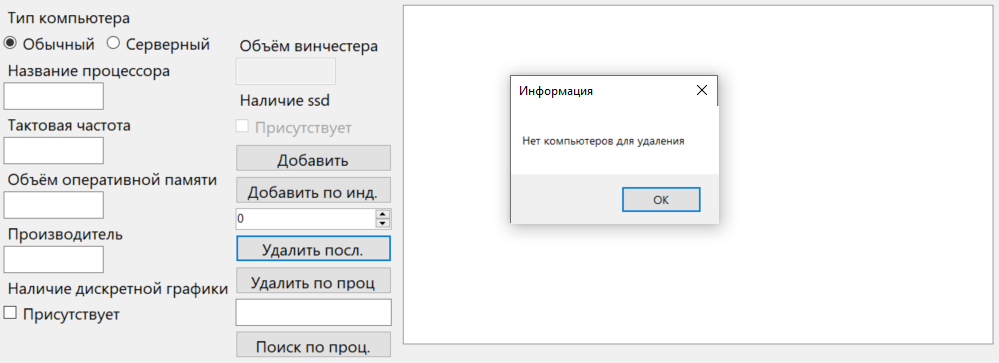
(рис. 27)

Ошибка при добавлении серверного компьютера без указания объёма HDD (рис. 28)



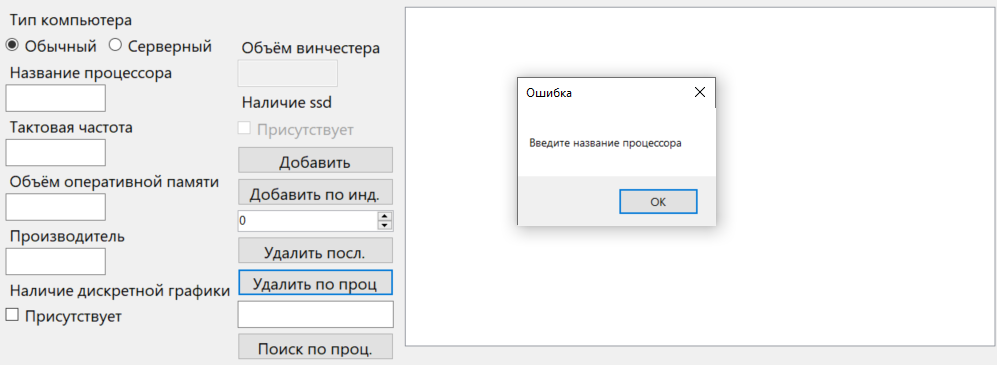
(рис. 28)

Ошибка при удалении последнего в пустом списке (рис. 29)



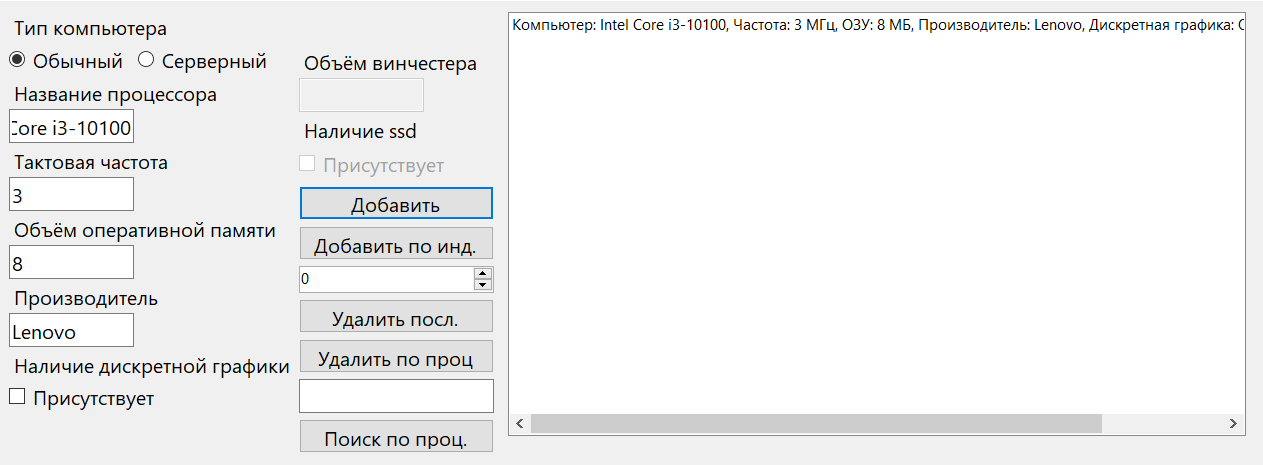
(рис. 29)

Ошибка при удалении по названию процессора при пустом поле ввода (рис. 30)



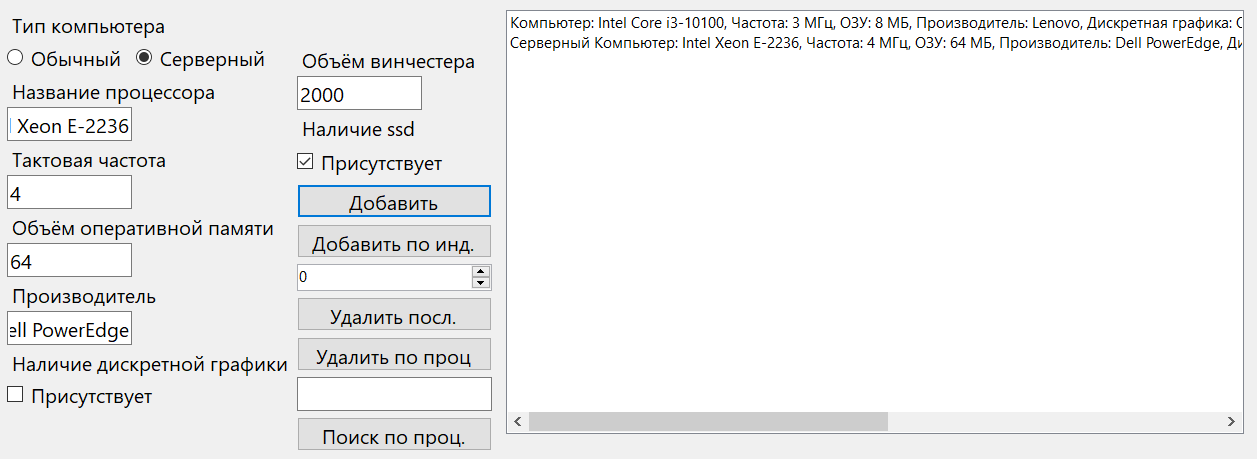
(рис. 30)

Добавление объекта обычного компьютера (рис. 31)



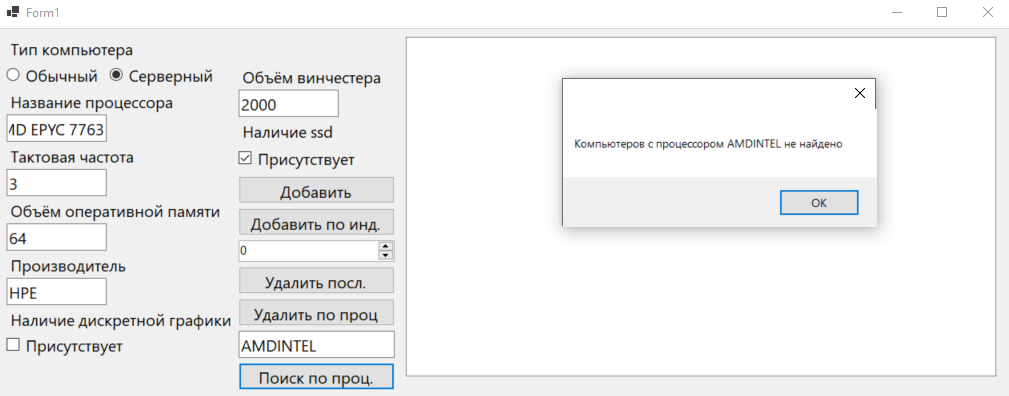
(рис. 31)

Добавление объекта серверного компьютера (рис. 32)



(рис. 32)

Ошибка при отсутствии процессора который ищем(рис. 33)



(рис. 33)

Поиск компьютеров по конкретному процессору (рис. 34)

(рис. 34)

