

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ**  
**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**  
**ИТМО»**  
**(Университет ИТМО)**

Факультет **Прикладной информатики**

Образовательная программа **Мобильные и сетевые технологии**

Направление подготовки(специальность) **09.03.03 Прикладная информатика**

**ПРАКТИЧЕСКАЯ РАБОТА №1**

**По дисциплине «Объектно-ориентированное  
программирование»**

**Тема "Вендинговый автомат"**

**Выполнил** Мкртчян К.Г.; К3241.

**Проверил** Слюсаренко С.В..

**Дата** 28.09.2025

## **Оглавление**

1. Общее описание
  2. Архитектура программы
  3. Классы и их функциональность
  4. Бизнес-логика
  5. Особенности реализации
  6. Примеры использования
  7. Вывод
- 

## **Общее описание**

Программа: Система управления торговым автоматом

Язык программирования: C#

Платформа: .NET 8.0

Тип приложения: Консольное приложение

Программа имитирует работу торгового автомата с двумя режимами:

- Режим покупателя - покупка товаров
- Режим администратора - управление товарами и денежными средствами

## **Архитектура программы**

### **Структура проекта:**

```
VendingMachine/  
└── Program.cs      // Точка входа  
└── Machine.cs     // Основной класс автомата  
└── Admin.cs        // Класс администратора  
└── VendingMachine.csproj
```

### **Диаграмма классов:**

Program → Machine → Admin

## **Классы и их функциональность**

### **1. Класс Machine - основной функционал автомата**

Поля:

- `_money` - словарь денежных средств (номинал → количество)
- `_products` - словарь товаров (ID → название)
- `_products_info` - информация о товарах (ID → [цена, количество])

Основные методы:

- `PrintOptions()` - главное меню выбора режима
- `PrintProductsList()` - отображение списка товаров
- `AcceptMoney()` - обработка оплаты
- `GiveChange()` - выдача сдачи
- `ContinueOperations()` - продолжение работы

### **2. Класс Admin - административные функции**

Поля:

- `_admins` - словарь логинов и паролов
- Наследует словари от Machine для работы с данными

Основные методы:

- `LogIn()` - аутентификация администратора
- `ProductOperations()` - управление товарами
- `MoneyOperations()` - управление денежными средствами
- `PrintOptionsList()` - меню администратора

### **3. Класс Program - точка входа**

Инициализация начальных данных и запуск автомата.

## **Бизнес-логика**

**Работа с денежными средствами:**

- Поддержка номиналов: 1, 2, 5, 10, 50, 100, 200, 500, 1000, 2000, 5000 рублей
- Ограничение хранения: не более 1000 единиц каждого номинала
- Алгоритм выдачи сдачи (жадный алгоритм)

### **Управление товарами:**

- Уникальные ID товаров
- Отслеживание цены и количества
- Валидация операций

### **Безопасность:**

- Базовая аутентификация администратора
- Проверка корректности вводимых данных

## **Особенности реализации**

### **1. Структуры данных:**

```
// Деньги: номинал → количество
Dictionary<ushort, ushort> _money;
```

```
// Товары: ID → название
Dictionary<ushort, string> _products;
```

```
// Информация: ID → [цена, количество]
```

```
Dictionary<ushort, List<ushort>> _products_info;
```

### **2. Обработка ввода:**

- Использование Convert.ToInt16() для числового ввода
- Рекурсивные вызовы при ошибках ввода
- Валидация данных на каждом этапе

### **3. Лингвистические особенности:**

```
// Правильное склонение слов для вывода  
public static string DeclineByCases(ushort count, ushort denomination)  
{  
    // Логика склонения "монета/монеты/монет"  
}
```

## Примеры использования

### **Сценарий 1: Покупка товара**

Выберите действие:

1. Приобрести товар
2. Добавить товар (+ денежные операции)

1

Список товаров в наличии:

1. Шоколад Белый, цена: 100 руб., кол-во: 3
2. Шоколад Горький, цена: 120 руб., кол-во: 4

Выберите товар: 1

Внесите сумму 100

### **Сценарий 2: Режим администратора**

Введите логин: admin

Введите пароль: qwerty

Вам разрешено:

1. Добавление, изменение и удаление продуктов
2. Снятие и внесение средств

## Вывод

Программа представляет собой полнофункциональную систему управления торговым автоматом с четким разделением ответственности между классами. Реализована базовая бизнес-логика, безопасность и удобство использования.