МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

кафедра Информационные системы

Бариев Эмин Юсуфович

Институт информационных технологий и управления в технических системах

курс 4 группа ИС/б-16-2

09.03.02 Информационные системы и технологии (уровень бакалавриата)

ОТЧЕТ

по лабораторной работе №2

По дисциплине: «Рефакторинг программного обеспечения»

По теме: «Рефакторинг программного кода. Перемещение функций между объектами»

|  |  |  |
| --- | --- | --- |
| Отметка о зачете | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_ |
|  |  | (дата) |
|  | Руководитель практикума |  |
| ст. преподаватель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Строганов В.А. |
| (должность) | (подпись) | (инициалы, фамилия) |

Севастополь 2019

**1 Цель работы**

Исследовать эффективность перемещения функций между объектами при рефакторинге программного кода. Получить практические навыков применения приемов рефакторинга объектно-ориентированных программ.

**2 Постановка задачи**

1. Выбрать фрагмент программного кода для рефакторинга.
2. Выполнить рефакторинг программного кода, применив не менее 7 приемов.
3. Составить отчет, содержащий подробное описание каждого модифицированного фрагмента программы и описание использованного метода рефакторинга.

**3 Ход работы**

3.1 Перемещение метода (Move Method)

На рисунке 1 выделен метод, который можно переместить в другой класс, где он вызывается.

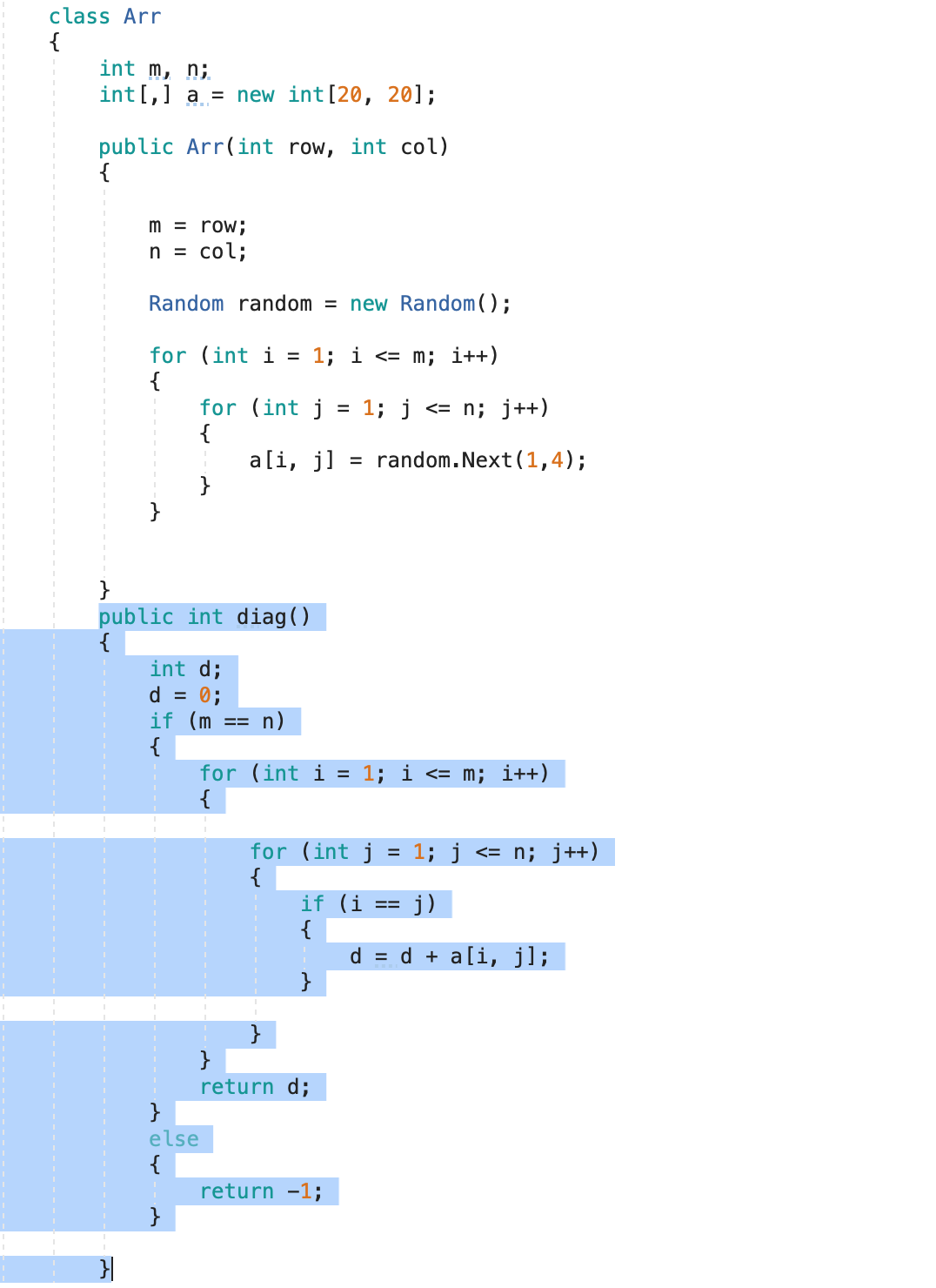


Рисунок 1 – программный код до перемещения метода



Рисунок 2 – программный код класса PerfectNum, который использует метод класса Arr

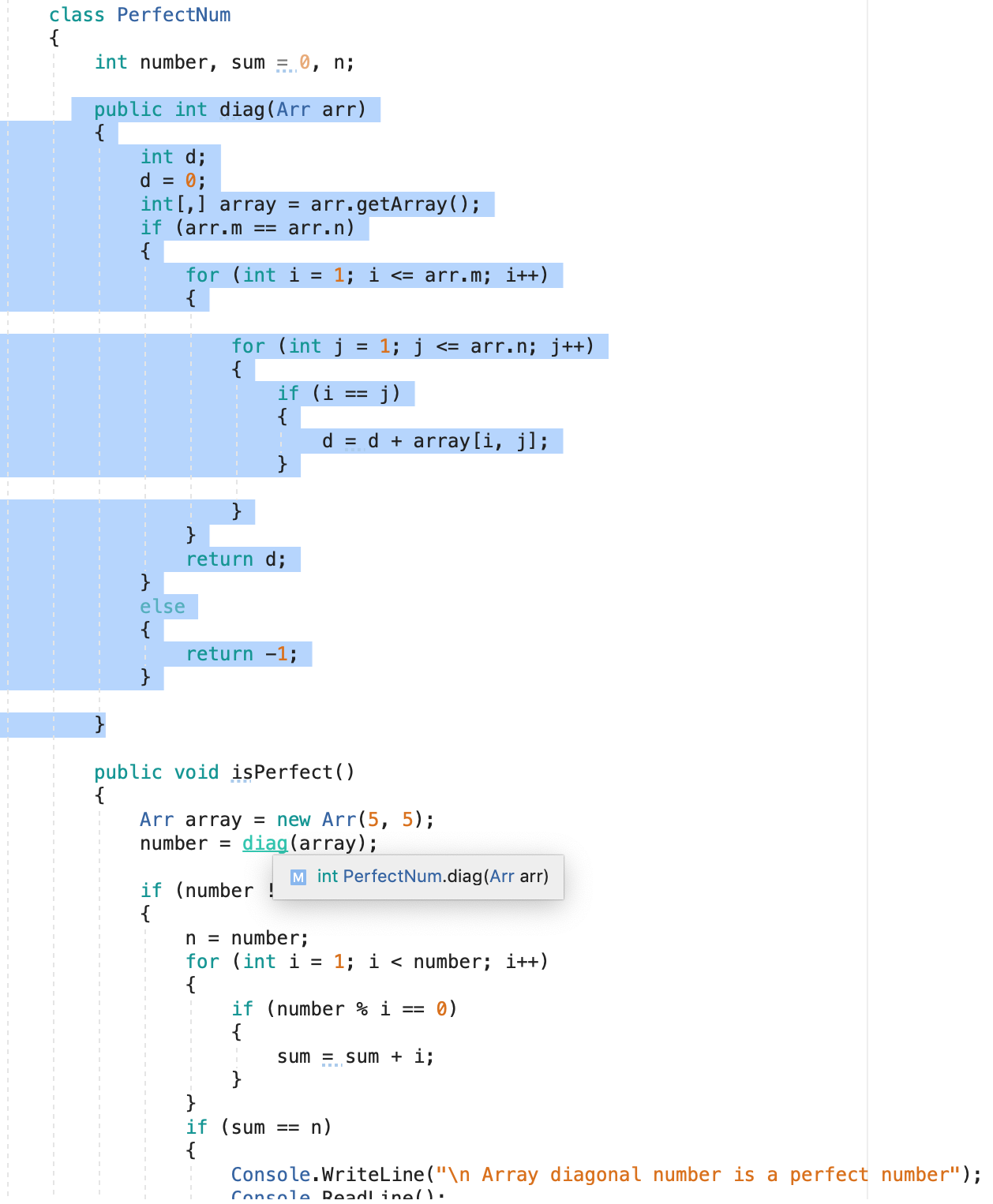


Рисунок 3 – программный код класса PerfectNum после перемещения туда метода из класса Arr

2. Извлечение класса (Extract Class)

На рисунке 4 показан программный код класса Arr, из которого можно выделить новый класс, который будет реализовывать функцию isPerfect().

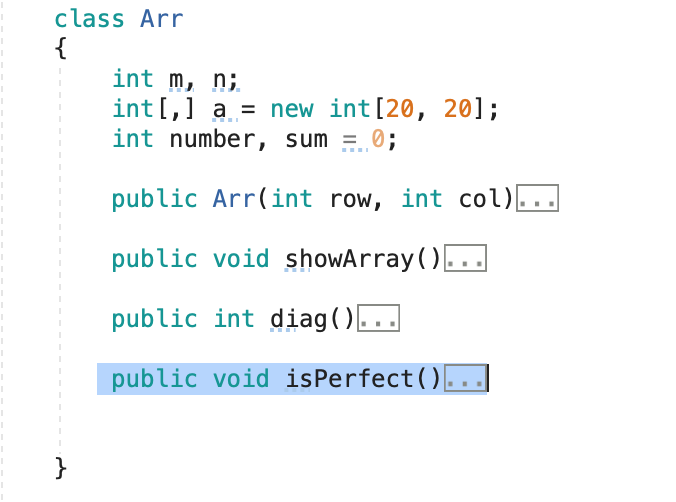


Рисунок 4 – программный код класса Arr до извлечения класса

На рисунке 5 показан программы код извлеченного класса PerfectNum.

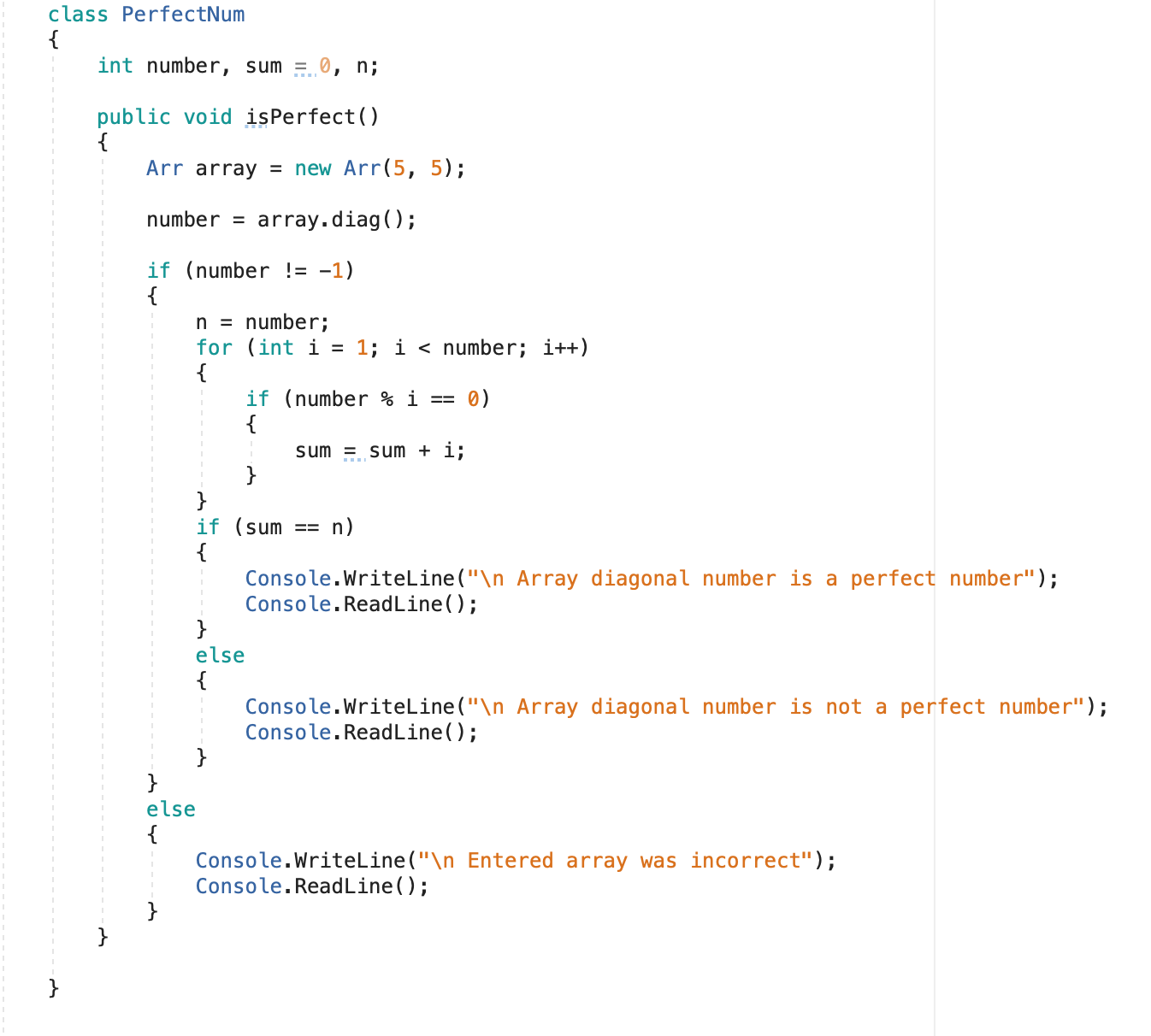


Рисунок 5 – программный код класса PerfectNum

3. Введение внешнего метода (Introduce Foreign Method)

Служебный класс не содержит нужный метод, при этом у нас нет возможности добавить метод в этот класс. Для этого добавим метод в клиентский класс и передадим в него объект класса в качестве аргумента.

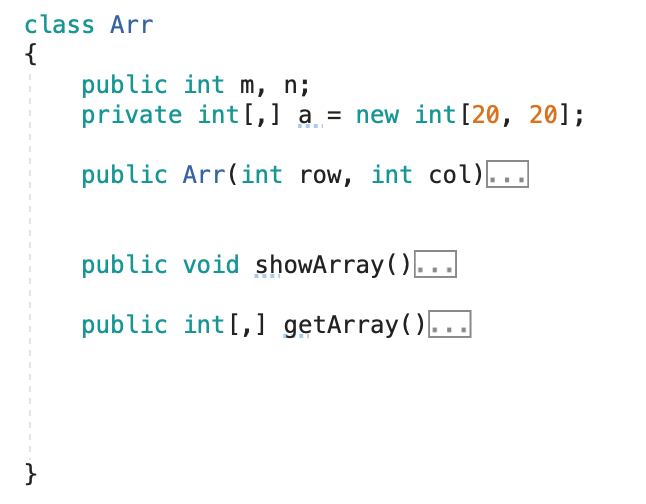


Рисунок 6 – программный код до введения внешнего метода

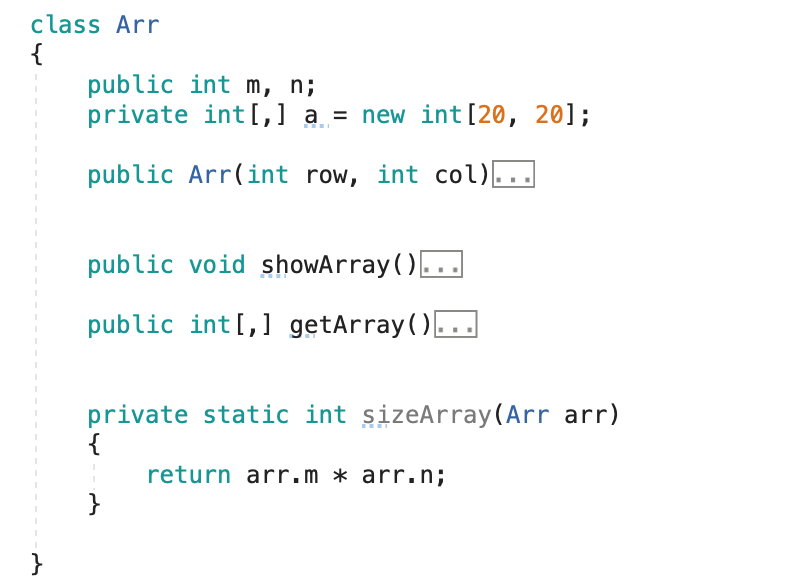


Рисунок 7 – программный код после введения внешнего метода

4. Введение локального расширения (Introduce Local Extension)

В служебном классе отсутствует нужные методы, при этом добавить их в этот класс не получается.

Создадим новый класс, который бы содержал эти методы, и сделаем его наследником служебного класса, либо его обёрткой.

На рисунке 8 показан программный код с введением локального расширения.

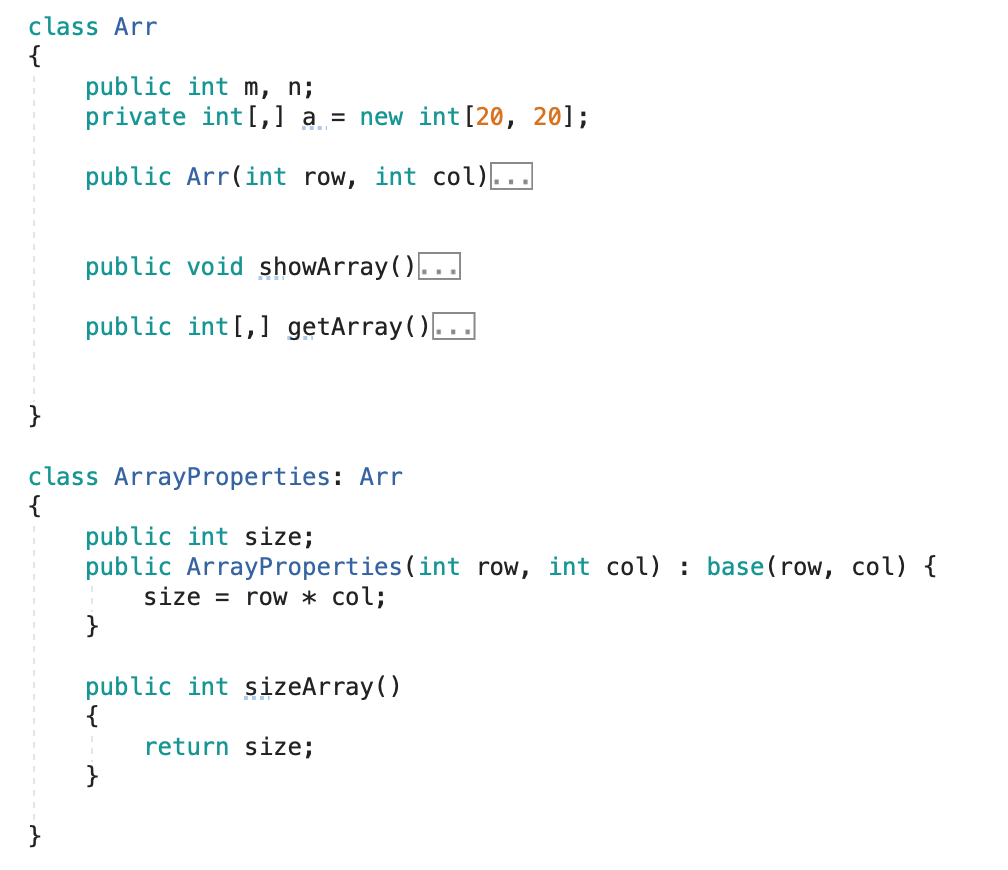


Рисунок 7 – программный код с введением локального расширения.

**Выводы**

В ходе выполнения лабораторной работы была исследована эффективность перемещения функций между объектами при рефакторинге программного кода. Так же были получены практические навыки применения приемов рефакторинга методов. При выполнении данной лабораторной работы были применены следующие техники рефакторинга:

* Перемещение метода (Move Method)
* Извлечение класса (Extract Class)
* Введение внешнего метода (Introduce Foreign Method)
* Введение локального расширения (Introduce Local Extension)