Лабораторная работа №1 МДК 01.01.

Гаврильченко Степан П-30

Вариант №2

№1 Описание программных функций

1. Создание списка с информацией о футболистах
2. Ввод запроса пользователя по названию команды
3. Поиск всех футболистов по названию команды
4. Сортировка футболистов команды по количеству забитых мячей
5. Поиск футболистов с наивысшим количеством мячей. Результат – список топ 3 футболиста команды с наивысшим количеством забитых мячей
6. Вычисление разницы баллов по сравнению с количеством баллов команды, занявшей первое место
7. Вывод на экран списка топ 3 игрока по количеству забитых мячей и разницу баллов по сравнению с командой, занявшей первое место

№2 Описание структуры информации

1. Сведения о футболисте:

String: Имя

Int: число забитых мячей

Class Player

1. О команде футболистов:

List<Player> Players;

String: название команды

Int: количество набранных очков

Class Team

1. Конструкторы для классов Player и Team, обозначение переменных

Public Player (string name, int goals)

{

Name = name;

Goals = goals;

}

Public Team (string name, int points, List<Player> players)

{

Name = name;

Points = points;

Players = players;

}

1. Функция показа топа бомбардиров

Метод GetTeams()

Метод создает и возвращает фиксированный набор команд List<Team> с предварительно заданными значениями. Каждая команда включает название, очки и список игроков с количеством забитых голов.

Метод FindTeam()

Метод проходит по переданному списку команд (List<Team>) и сравнивает названия каждой команды с введённым именем, игнорируя регистр символов. Если совпадение найдено, метод немедленно возвращает найденную команду. Если команда не найдена, возвращается null.

Метод GetTopScorers предназначен для нахождения трёх лучших бомбардиров среди списка игроков (List<Player>)

1. Инициализируется пустой список для хранения лучших бомбардиров.
2. Для каждого игрока из переданного списка выполняются следующие шаги:

* Если в списке лучших бомбардиров еще меньше трех элементов, игрок добавляется в список.
* Иначе, для каждого существующего бомбардира проверяется, забил ли новый игрок больше голов. Если да, он заменяет одного из существующих бомбардиров.

1. Если после проверки очередного игрока в списке становится больше трех бомбардиров, удаляется тот, кто забил наименьшее количество голов.
2. После обработки всех игроков возвращается итоговый список из трех лучших бомбардиров.

Метод PrintTopScorers()

Метод последовательно проходит по списку лучших бомбардиров (List<Player> topScorers) и печатает их имена и количество забитых голов на консоли.

Метод GetMaxPoints()

Метод проходит по списку команд (List<Team>) и отслеживает команду с наибольшим количеством очков. Максимальное значение сохраняется и возвращается в конце.