

Весь алгоритм задания можно описать в 4 действиях:

```
namespace RadioEntryTask
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            Contest contest = new Contest()
                .EnterNumOfCandidates()
                .FillListOfCandidates()
                .SortListOfCandidates()
                .CreateTeamsAndPrint();
        }
    }
}
```

ввод данных,

сортировка,

создание команд,

вывод на экран.

Ничего лишнего в Main() нет.

Класс Candidate хранит информацию о каждом кандидате: его номер, группу, очки и тип (руководитель или обычный игрок).

```
namespace RadioEntryTask
{
    4 references
    public enum TypePerson { lead, common }
    4 references
    public class Candidate(int id, int groupNum, int[] roundScore)
    {
        public int candidateNumber = id;
        public int groupNumber = groupNum;
        public int[] roundScore = roundScore;
        public int globalScore = roundScore.Sum();
        public TypePerson typePerson;
    }
}
```

Я использовал enum вместо bool, так как если вдруг появятся новые типы игроков, их можно будет легко добавить.

Класс Contest является сердцем программы. Тут написана все методы и логика.

```
namespace RadioEntryTask
{
    6 references
    public class Contest
    {
        2 references
        public byte k { get; private set; }
        private List<Candidate> candidates = new List<Candidate>();
        public int[] team1, team2;

        1 reference
        public Contest EnterNumOfCandidates()
        {
            k = byte.Parse(Console.ReadLine());
            return this;
        }
    }
}
```

Почему тут возвращается Contest? Это называется FluentApi, стиль вызова публичных методов.

Когда методы вызываются по цепочке. Ниже приведен пример.

FLUENT
API

```
0 references
static void Main(string[] args)
{
    Contest contest = new Contest()
        .EnterNumOfCandidates()
        .FillListOfCandidates()
        .SortListOfCandidates()
        .CreateTeamsAndPrint();
}
```

```
0 references
static void Main(string[] args)
{
    Contest contest = new Contest();
    contest.EnterNumOfCandidates();
    contest.FillListOfCandidates();
    contest.SortListOfCandidates();
    contest.CreateTeamsAndPrint();
}
```

~~FLUENT
API~~