

Домашняя работа №12
22.02.2022

Вам необходимо смоделировать чемпионат по футболу среди нескольких команд, причем каждый день играет ровно одна пара команд. За победу присуждается 3 очка, за ничью — 1 очко, за проигрыш — 0 очков. Необходимо поддерживать таблицу результатов чемпионата в актуальном состоянии и отвечать на запросы: общая таблица, лучшая команда, общее количество игр и голов.

Класс «команда»:

```
class Team{
    char name[10];    //название команды
    int wins;         //число побед
    int lose;         //число проигрышей
    int draw;         //число ничьих
    int goalScored;   //число забитых голов
    int goalConceded; //число пропущенных голов
    int points();
public:
    Team();
    Team(const Team & someTeam);
    void setName(const char * newName);
    void addResult(int scored, int conceded);
    bool is(const char * someName);
    int compare(const Team & team);
    void print();
    int games();
    int goals();
};
```

Необходимо описать:

1. `int points()` — метод для подсчета количества очков;
2. `Team()` — конструктор по умолчанию (инициализирует все поля);
3. `Team(const Team & someTeam)` — конструктор копирования (обратите внимание, что необходимо ГЛУБОКОЕ копирование, так как поле `name` является указателем, копировать все символы можно функцией `char *strcpy(char *dest, const char *src)` из библиотеки `<cstring>`);
4. `void setName(const char * newName)` — метод для задания (сеттер) названия команды;
5. `void addResult(int scored, int conceded)` — метод для добавления результатов игры (обновить соответствующие внутренние поля);
6. `bool is(const char * someName)` — метод для проверки соответствия названия;

7. `int compare(const Team & team)` метод для сравнения с результатом другой команды (если данная команда лучше, то вернуть 1, если они выступили одинаково, то вернуть 0, если другая команда лучше, то вернуть -1). Сравнение 2 команд производится в приоритете: по очкам; если очки равны, то по разнице забитых и пропущенных мячей; если разница одинакова, то по количеству забитых; если количество забитых одинаково, то команды равны.
8. `void print()` — метод для печати строки результатов (название, число игр, число выигрышей, число ничьих, число проигрышей, число забитых голов, число пропущенных голов, очки). Для красивого вывода пользуйтесь манипулятором `std::setw()` из библиотеки `<iomanip>`;
9. `int games()` — метод для подсчета количества игр;
10. `int goals()` — метод для подсчета голов.

Класс «таблица»:

```
class Table{
    int teamNumber;    //количество команд
    Team name[10];     //команды
    void printHeader();
public:
    Table();
    void addResult(const char * team1, const char * team2, int goal1, int goal2);
    void print();
    void printBest();
    void printStat();
};
```

Необходимо описать:

1. `void printHeader()`; — метод для печати строки шапки (название, число игр, число выигрывшей, число ничьих, число проигрывшей, число забитых голов, число пропущенных голов, очки). Для красивого вывода пользуйтесь манипулятором `std::setw()` из библиотеки `<iomanip>`.
2. `Table()` — конструктор (инициализирует поле);
3. `void addResult(const char * team1, const char * team2, int goal1, int goal2)` — метод для добавления результатов игры. Если хотя бы один из параметров, соответствующий числу голов, отрицательный или название команды содержит более 9 символов, то вывести сообщение о некорректности и ничего не добавлять, иначе — добавить результаты к соответствующим командам. Если команда есть в списке (проверяется с помощью метода `bool Team::is()`), то добавить результаты с помощью `void Team::addResult(int scored, int conceded)`; Вызывается при команде «+ team1 team2 goal1 goal2».
4. `void print()` — метод для печати результатов всех команд с помощью метода `void Team::print()`. Вызывается при команде «?».
5. `void printBest()` — метод для поиска и печати результатов лучшей команды. Если команд нет, то сообщить об этом, если команды есть, то скопировать первую команду с помощью конструктора копирования `Team::Team(const Team &)` и найти лучшую стандартным алгоритмом поиска максимума, сравнивая с другими командами методом `int Team::compare(const Team & team)`. Если лучших команд несколько то печатать все. Вызывается при команде «*».
6. `void printStat()`; — метод для печати общего количества игр и голов (можно посчитать с помощью методов `int Team::goals()` и `int Team::games()`). Вызывается при команде «s».

Ввод

Вывод

```

|+ birds pigs 2 4 |
|+ angry pigs 3 3 |
|?                |team      games wins  draws loses  goals goalc points
|                  |birds      1      0      0      1      2      4      0
|                  |pigs       2      1      1      0      7      5      4
|                  |angry      1      0      1      0      3      3      1
|s                |games: 2
|                  |goals: 12
|+ eagle angry 5 1 |
|?                |team      games wins  draws loses  goals goalc points
|                  |birds      1      0      0      1      2      4      0
|                  |pigs       2      1      1      0      7      5      4
|                  |angry      2      0      1      1      4      8      1
|                  |eagle      1      1      0      0      5      1      3
|*                |bestteam  games wins  draws loses  goals goalc points
|                  |pigs       2      1      1      0      7      5      4
|.                |

```