

宮城県将棋連盟からの依頼

対戦表を作成するプログラム 仕様

1、どこで使うか

(ア)大会の組み分けに使用する

(イ)参加者は、10～30 人に仮定する

2、現在の利用方法

(ア)参加者は大会会場に到着

(イ)参加者は参加登録を行う

① 名前、くじ番号を申請

② 遠方から一緒に来たなど、「できるだけ対戦相手にならないようにしてほしい」
人たち（チーム）を申請

(ウ)参加者が奇数の場合、「不戦敗」となる参加者を作成する

(エ)主催者は1回目の対戦表を作成する

① 同じチームの人とは、できるだけ対戦しないようにする

② くじの番号の若いほうから順番に対戦表を作成する

(オ)参加者は対戦し、主催者に結果を報告する

① 主催者は、勝敗を記録する

(カ)主催者は2回目の対戦表を作成する

① 最優先：今日既に対戦している人と対戦しないこと

② 優先：同じ勝数の人と対戦すること。同じ勝ち数の人が奇数の場合、勝ち数が
1 少ない人を1人入れる。

③ できるだけ：同じチームの人と対戦しないようにしたいが、仕方なければ対戦
してもよい

(キ) (オ) と (カ) を繰り返す

(ク)点数計算

① それぞれの参加者が、何勝したか計算する

② それぞれの参加者の、それぞれの試合ごとに、勝った場合は相手の勝数を加算
する（強い相手に勝った人の点数を高くする）

C#でプログラムを作成する

1、提供されるクラスについて

(ア) Person クラス

- ① 参加者について登録するクラス
 - 1. LotNumber・・・くじの番号
 - 2. PersonGroup・・・グループ（アルファベット大文字 1 文字）
 - 3. PersonName・・・参加者名

(イ) Match クラス

- ① 対戦する Person のくじ番号と、勝敗（または結果が出ていない）を記録
- ② コンストラクタ（完成）
 - 1. くじ番号を 2 つ渡す。Result=0（結果が出ていない）を設定する。
- ③ MakeMatch メソッド（未完成）
 - 1. 第 1 引数：くじ番号を Key に、Person を Value に設定した Dictionary
 - 2. 第 2 引数：今までの対戦が格納された Match の配列
 - 3. 戻り値：対戦の一覧
- ④ MergeMatch メソッド（未完成）
 - 1. 第 1 引数：今までの対戦が格納された Match の配列
 - 2. 第 2 引数：今回の対戦が格納された Match の配列
 - 3. 戻り値：第 1 引数に第 2 引数を結合させた Match の配列

(ウ) Score クラス（未定義）

- ① くじ番号と、勝数、最終得点を記録
- ② CalcWinCount メソッド（未定義）
 - 1. 第 1 引数：最終戦の結果まで含めた Match の配列
 - 2. 戻り値：くじ番号を Key に、Score のインスタンスを Value に設定した Dictionary<int,Score> ※最終得点は 0 に設定する
- ③ CalcScore メソッド（未定義）
 - 1. 第 1 引数：最終戦の結果まで含めた Match の配列
 - 2. 戻り値：くじ番号を Key に、Score のインスタンスを Value に設定した Dictionary<int,Score> ※最終得点も格納する

それ以外に、メソッドの中で必要な処理があれば、それぞれ作成し、テストを追加して構わない。