

3. This question involves pairing competitors in a tournament into one-on-one matches for one round of the tournament. For example, in a chess tournament, the competitors are the individual chess players. A game of chess involving two players is a match. The winner of each match goes on to a match in the next round of the tournament. Since half of the players are eliminated in each round of the tournament, there is eventually a final round consisting of one match and two competitors. The winner of that match is considered the winner of the tournament.

Competitors, matches, and rounds of the tournament are represented by the `Competitor`, `Match`, and `Round` classes. You will write the constructor and one method of the `Round` class.

```
/** A single competitor in the tournament */
public class Competitor
{
    /** The competitor's name and rank */
    private String name;
    private int rank;

    /**
     * Assigns n to name and initialRank to rank
     * Precondition: initialRank >= 1
     */
    public Competitor(String n, int initialRank)
    { /* implementation not shown */ }

    /* There may be instance variables, constructors,
       and methods that are not shown. */
}
```

```
/** A match between two competitors */
public class Match
{
    public Match(Competitor one, Competitor two)
    { /* implementation not shown */ }

    /* There may be instance variables, constructors,
       and methods that are not shown. */
}

/** A single round of the tournament */
public class Round
{
    /** The list of competitors participating in this round */
    private ArrayList<Competitor> competitorList;

    /** Initializes competitorList, as described in part (a) */
    public Round(String[] names)
    { /* to be implemented in part (a) */ }

    /**
     * Creates an ArrayList of Match objects for the next round
     * of the tournament, as described in part (b)
     * Preconditions: competitorList contains at least one element.
     *                 competitorList is ordered from best to worst rank.
     * Postcondition: competitorList is unchanged.
     */
    public ArrayList<Match> buildMatches()
    { /* to be implemented in part (b) */ }

    /* There may be instance variables, constructors,
       and methods that are not shown. */
}
```