Planning Step 1

Requirements

1. Tracks games played and their outcome (who won).
2. Multiple competitors play in the tournament.
3. Creates a tournament plan (who plays in what order).
4. Schedules games.
5. A single loss eliminates a player.
6. The last player standing is the winner.

Questions

1. How many players will the tournament handle? Is it variable?  
   The application should be able to handle a variable number of players in a tournament.
2. If a tournament has less than the full complement of players, how do we handle it?   
   A tournament with less than the perfect number should add in “byes”. Basically, certain people selected at random get to skip the first round and act as if they won. Random assignment for “byes”.
3. Should the ordering of who plays each other be random or ordered by input order?

The ordering of the tournament should be random.

1. Should we schedule the game or are they just played whenever?  
   The games should be played in whatever order and whenever the players want to play them.
2. If the games are scheduled, how does the system know when to schedule games for?  
   They are not scheduled so we do not care.
3. If the games are played whenever, can a game from the second round be played before the first round is complete?  
   No. Each round should be fully completed before the next round is displayed.
4. Does the system need to store a score of some kind or just who won?  
   Storing a simple score would be nice. Just a number for each player. That way, the tracker can be flexible enough to handle a checkers tournament (the winner would have a 1 and the loser a 0) or a basketball tournament.
5. What type of frond-end should this system have (form, webpage, app, etc.)?  
   The system should be a desktop system for now, but down the road we might want to turn it into an app or a website.
6. Where will the data be stored?  
   Ideally, the data should be stored in a Microsoft SQL database but please put in an option to store to a text file instead.
7. Will this system handle entry fees, prizes, or other payouts?  
   Yes. The tournament should have the option of charging an entry fee. Prizes should also be an option, where the tournament administrator chooses how much money to award a variable number of places. The total cash amount should not exceed the income from the tournament. A percentage-based system would also be nice to specify.
8. What type of reporting is needed?  
   A simple report specifying the outcome of the games per round as well as a report that specifies who won and how much they won. These can be just displayed on a form or they can be emailed to tournament competitors and the administrator.
9. Who can full in the results of a game?

Anyone using the application should be able to fill in the game scores.

1. Are there varying levels of access?  
   No. The only method of varied access is if the competitors are not allowed into the application and instead, they do everything via email.
2. Should this system contact users about upcoming games?

Yes, the system should email users that they are due to play in a round as well as who they are scheduled to play.

1. Is each player on their own or can teams use this tournament tracker?

The tournament tracker should be able to handle the addition of other members. All members should be treated as equals in that they all get tournament emails. Teams should also be able to name their team.

Big Picture Design

* Structure: Windows Forms application and Class Library
* Data: SQL and/or Text File
* Users: One at a time on one application
* Key concepts:
  + Email
  + SQL
  + Custom Events
  + Error Handling
  + Interfaces
  + Random Ordering
  + Texting

Data Design - Mapping the Data

Team

* TeamMembers (List<Person>)
* TeamName (string)

Person

* FirstName (string)
* LastName (string)
* EmailAddress (string)
* CellphoneNumber (string)

Tournament

* TournamentName (string)
* EntryFee (decimal)
* EnteredTeams (List<Team>)
* Prizes (List<Prize>)
* Rounds (List<List<Matchup>>)

Prize

* PlaceNumber (int)
* PlaceName (string)
* PrizeAmount (decimal)
* PrizePercentage (double)

Matchup

* Entries (List<MatchupEntry>)
* Winner (Team)
* MatchupRound (int)

MatchupEntry

* TeamCompeting (Team)
* Score (double)
* ParentMatchup (Matchup)

User Interface Design

1. Tournament Viewer - Form
2. Create Tournament - Form
3. Create Team – Form
4. Create Prize – Form
5. Tournament Dashboard

Logic Planning

* Tell how do the buttons work in the Forms
* Data Access, Data Stored, Email out – trigger point

Class Library Creation