

---

---

# COMS W4115 Programming Languages and Translators

---

---

(Professor: Stephen A. Edwards)

## Project Proposal: Fantasy Football Stat Tracker Compiler

Michael Lam  
Email: [michael.lam@lmco.com](mailto:michael.lam@lmco.com)  
Phone: 215-815-5629  
Due Date: 9/25/2007

## Table of Contents

1	Description.....	3
2	Code Snippet #1.....	4
3	Code Snippet #2.....	4
4	Code Snippet #3.....	4

# 1 Description

My inspiration for designing a Fantasy Football Stat Tracker compiler transpired while participating in a fantasy football league with some friends. While the cost for participating in the fantasy football league is free of charge, most of the online fantasy football services charge a bundle for using their online stat tracker. Considering the circumstances for not willing pay for something when I obtain it for free, this is my opportunity to design a compiler for Stat Tracking.

The features of the Fantasy Football Stat Tracker compiler will contain the following:

- Sorting algorithm for player stats
- Filtering for position by offense, defense, or specific position.
- Stats comparison by previous week or previous year
- Search function for players in the database
- Semantic checks for errors
  - Adding receiving stats to a defensive player will cause an error.
  - Adding field goal stats to a quarterback will be invalid.
  - Adding 1-point conversion stat to a quarterback will be invalid; however, a 2-point conversion will be valid.

## Keywords:

- sort
- retrieve
- report
- compare
- search

## Comments:

@This is a comment@

## Data Type Identifiers:

These data types represent various types of scoring

- QB
- RB
- WR
- KKR
- TE
- DEF
- DPLY
- rshYards
- rcvYards
- passYards
- passINT
- rshTD
- rcvTD
- passTD
- fumLost

## 2 Code Snippet #1

```
QB myStartingQB;  
RB myStartingRB;
```

```
rshYards QBrushYards;  
passYards QBPassYards;  
passTD QBPassTD;
```

```
@ Retrieve to stats for Bret Favre@  
myStartingQB {retrieve: "Favre"};
```

```
@ Print the stats for my starting quarterback. Option 1 prints all statistical categories @  
myStartingQB {report: 1, 2006};
```

## 3 Code Snippet #2

```
@ Sort by most rushing yards and return top 10, stored in array tempRB@  
tempRB[10] = myRB {sort: RBrushYards, 10}
```

## 4 Code Snippet #3

```
RB betterRB;
```

```
@ Compare the 2006 stats for two running backs and return the running back with better  
stats@  
betterRB = {compare: 2006, "Westbrook", "Alexander"};
```