MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS

CNG140 C Programming – Project 6

Date handed out: **Thursday 17 May 2012**

Date submission due: **Saturday 26 May 2012**

"12 Dev Adam"

This assignment familiarizes you with string arrays, structures, dynamic memory allocation, and files. In doing so, you will summarize the performance of the Turkish National Basketball players, aka "12 Dev Adam". This project consists of two parts.

PART I: Reading and Displaying Data

The first part of your program will read some player information from a file, format the information, put it into an array of structures, and then display that array on the screen. You will find the following structure useful:

```
struct playerInfo
{
   char playerName[25];
   char position[2];
   int freethrowscored;
   int freethrowshots;
   int twopointsscored;
   int twopointsshots;
   int threepointsscored;
   int threepointsshots;
};
```

The input for your program is in a file called **basketball.txt**. Each record in the file represents one player and is made up of multiple fields that are separated by colons. The fields for each player are: the player name, position, free throws scored, free throws shots, two points scored, two points shots, three points scored, and three points shots. A record in the file resembles the following:

```
Cenk Akyol:f:20:40:30:60:40:80
```

Your main function should open **basketball.txt**, check that it opened correctly, and should invoke a function (**readPlayer**) that reads the contents of basketball.txt into an array of playerInfo structures. The readPlayer function should take as input the array of playerInfo structures and it should return the number of players in the array. Upon return of readPlayer, your program should invoke **displayPlayer**, a new function, which takes the playerInfo array and the number of elements in this array, and it displays for each player, their name, as follows:

```
            12 Dev Adam 2011-2012 Player Statistics

Player               Points scored   Free Throw%  TwoPoints%   ThreePoints%
-----------------------------------------------------------------------
Cenk Akyol             200            50.0           50.0      50.0
```

…

Use the following formulas for the calculated values (points scored, free throw%, two points% and three points%):

```
Points scored = freethrowscored*1 + twopointsscored*2 + threepointsscored*3;

Free throw% = freethrowsscored / freethrowshots * 100
Two points% = twopointsscored / twopointsshots * 100
Three points% = threepointsscored / threepointsshots * 100
```

PART II: Saving Data:

Extend your program to include a new function, **savePlayer**, that takes the array of playerInfo and the number of elements in this array and output the same table above to a file called **stats-12-dev-adam.txt**. Your function should check that the file opened correctly, and then output player information to this file.

Hint:

Use the ***strchr*** function to search for a colon (':') so that each field can be isolated, ***strcpy*** or ***atoi*** can then be used to put the isolated string into the appropriate field of the structure. The ***atoi*** function will be used to convert a string to its numeric representation.

Grading:

Your program will be graded as follows:

Grading Point	**Mark (10)**
Main function that opens the file and checks for error, invokes readPlayer, displayPlayer, and savePlayer, that reads arrays and displays their contents	2 points
readPlayer receives array by reference, fills array, and returns number of elements in array. It dynamically allocates memory for the array as it grows.	4 points
displayPlayer receives array by reference and the number of elements by copy, and displays the table on the screen	2 points
savePlayer receives array by reference and the number of elements by copy, and saves the table into teams.txt	2 points