Notes on Computer Project #10

Comments about the assignment and responses to frequently asked questions will be added to this file as necessary.

***** comments added on 11/05/18 *****

1. As stated on the assignment handout, you are required to create a makefile which controls the translation of your program, and the name of your executable files must be "proj10".

Since you must link the instructor-supplied driver module with your support module, your makefile should include the appropriate steps to assemble your version of "proj10.support.s", as well as link the object code modules.

2. Please note the following suggestion:

  You may wish to create a stub for the required function, then translate, link
  and execute the program to explore the behavior of the driver module.

3. A couple of definitions about hockey statistics:

A player’s points is defined as the sum of that player’s goals and assists.

A player’s points per game (a real number) is defined as number of points divided by number of games played. A player who has not participated in any games is defined to have zero points per game.

4. Be sure that you have the correct layout of both "struct player" and "struct table" in memory. For example, you must account for any bytes of padding to keep certain fields aligned on the appropriate byte boundary in memory.

5. Note that your program must work correctly for any properly formatted data set. An example of a properly formatted data file is available as:

```
  /user/cse320/Projects/project10.data
```

Clearly, you will need to develop several data files to test various aspects of your solution, since the simple instructor-supplied data file will not be sufficient for all of the cases that you will want to test.

6. Please note that the "r" option in the instructor-supplied driver module does not rely on any of your functions to perform its work.

7. Please note the function prototype below:

```
  int search( struct table*, unsigned long, struct player** );
```

The third argument to function "search" is a pointer to a pointer (an address where an address can be stored by the function).

Any function which calls function "search" is responsible for allocating four bytes of memory and sending the address of that four-byte area of memory as the third argument to "search".

Those four bytes can be allocated in the ".data" section (by a ".word" or ".skip" instruction) or in the run-time stack (by the "save" instruction which is the prologue for that function).

--M. McCullen