Jaxon Jensen

Project 2 Assembly break down

Computer Architecture

To start off the asm file is basically a direct translation of the .java file with the necessary attributes added that assembly needs to run properly. If anything is not clear referring to the java file would probably clear it up.

Generate: Uses the registers listed with their corresponding purpose in the comment above the function. It generates a random number, checks if it is already in the array using find, if it isn't it is then added to the array. If it does already exist it just repeats with a new random number until all three are generated. (Note: the stack only contains the $RA as the other used variables are always reset before usage throughout the code to ensure they are the proper values. This is also the case for findBalls and findStrikes.)

findBalls: Uses the registers listed with their corresponding purpose in the comment above the function. Takes the input array and uses find to see if that entry exists in generated. If it does it checks that it is not a strike (same index) and so long as it isn't a strike the ball counter gets incremented and after all three are checked it is returned

findStrikes: Uses the registers listed with their corresponding purpose in the comment above the function. Does the exact same process as findBalls but the opposite. If the entry is found at the same index the counter is incremented. The counter is returned after checking all three values.

Main:

1) Calls generate to create the array the player will guess
2) Prints generated for debugging purposes
3) Enters guessLoop where the game is played at
4) User is prompted for a guess which is read as a three-character string (EXTRA CREDIT)
5) Input string is parsed to ensure that only the characters 1-9 exist and there are no duplicates. If any other characters or duplicates are present, it will reprompt the user with the corresponding message.
6) At the same time as the check above is being done the numerical values are inserted into the guess array given they pass
7) FindStrikes is called and using the result the corresponding print is made
8) FindBalls is called and using the result the corresponding print is made
9) An additional check is done if findBalls returns zero where it checks if there are no strikes, and in that case, it prints "out"
10) If at step 7 the findStrikes function returns three (meaning a win) then the program jumps to the end where it asks the user if they want to play again. If the player wishes to play again, they must type 'Y' as all other character entries will result in program termination.