Progress Report

* Asks the user to input a set of players (max 6). For each player the user has to select a type (Elf, Human, Ogre, Wizard) and input a name.
* Each player will be represented as a struct characterised by the fields identifying the player (player type and player name), life points, and the fields characterising the player capabilities (Smartness, Strength, Magic Skills, Luck and Dexterity). The life points are represented as an integer initially set to 100. The capabilities are represented as integers that can assume values in [0, 100] and are randomly assigned to each player following the criteria indicated in section B.
* Subsequently the program asks the user to select the number of slots (max 20).
* The type of the slot is selected randomly. The type of a slot could be: Level Ground, Hill or City. Slots are stored in an array.
* Subsequently the players are placed in a slot randomly. Note that only one player can be assigned to a slot.
* After all the slots are created, the program asks each users whether s/he wants to move to the next or the previous slot - if possible - or whether s/he wants to attack the closest player.
* If a player decides to move to a slot it loses or gains capabilities according to the criteria described in Section C.
* If a player decides to perform an attack and there are two closest players, s/he should be able to choose between the two of them. The attacked and attacker players can lose or gain points according to the criteria described in Section D.
* After each player has chosen an action, the program prints the name and type of each player and its life points and terminates. These should be formatted as follows: <Player Name>(<Player Type>, <Life Points>. For example: