

```

1  #include "decl.h"
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <string.h>
5
6  /**
7   * Allocates a new PlayerInfo struct and assigns fields
8   * @param name The player's name
9   * @param date The current date
10  * @param gen The number of generations
11  * @return a new PlayerInfo struct
12  */
13  struct PlayerInfo *createPlayerInfo(char *name, char *date, int gen)
14  {
15      struct PlayerInfo *info = malloc(sizeof *info);
16      info->name = name;
17      info->date = date;
18      info->numGenerations = gen;
19      return info;
20  }
21
22  /**
23   * Returns a line of user input from the console without overflow
24   * @return A string containing the input
25   */
26  static char *getLine()
27  {
28      char *buffer = malloc(BUFFER_SIZE);
29      char temp;
30      int len = 0;
31      while ((temp = getchar()) != '\n' && len < BUFFER_SIZE - 1)
32      {
33          buffer[len++] = temp;
34      }
35      buffer[len] = '\0';
36      return buffer;
37  }
38
39  /**
40   * Prompts the player to enter pertinent info
41   */
42  void playerInfo()
43  {
44      char *name, *date, numBuffer[BUFFER_SIZE];
45      int numGenerations = 0;
46      bool isValid = FALSE;
47
48      printf("Enter Your Name >> ");
49      name = getLine();
50
51      printf("Enter Today's Date >> ");
52      date = getLine();
53
54      printf("Enter The Desired Number of Generations >> ");
55      scanf("%s", numBuffer);
56      isValid = sscanf(numBuffer, "%d", &numGenerations);
57
58      /*Prompt the user for valid input if necesssary*/
59      while (!isValid) {
60          printf("Please enter a valid Integer >> ");
61          scanf("%s", numBuffer);
62          isValid = sscanf(numBuffer, "%d", &numGenerations);
63      }
64
65      struct PlayerInfo *info = createPlayerInfo(name, date, numGenerations);
66      boardInitialization(info);
67      readPlayersInformation(info);
68      free(name);
69      free(date);

```

