Ain Shams University
Computer Engineering Department

Intelligent Mancala Game Report

CSE 481

Amir Abozaid Abdelaziz	1600308
Muhammed Muhammed Kamal	1601267
Ahmed Ehab Saad	1600043
Omar Samir Abu El-Azm	1600879
Fares Osama Salem	1600965

- **Brief description**

In this implementation, the user has the freedom to choose to be either first or second player, he has the ability to decide stealing or no-stealing mode, he controls the difficulty level of the game to be easy, medium, or hard (bonus feature), and finally he can change the default time limit of the CPU from the default which is 5 seconds to any number of seconds he wants (bonus feature).

- **Detailed description of the utility functions**

1- int stoi(string input_string) which converts the only parameter which is a string to its corresponding integer, but if the parameter contains any digit that does not represent number digit (from '0' to '9') it will be terminated, so in our code we first check if the input_string contains only number digits, and if so it will be passed directly to stoi(), if not , a warning message will be printed to the user asking him to enter only number digits and repeat this process again and again until he enters a valid string containing only number digits, this function is used to receive the selected timeout from the user if he wants to change it and not use the default (5 seconds) and also used to receive the move the user wants to play in his turn, to use this function we must include <string>

2- void Sleep(int milliseconds) which forces the whole program to be delayed to the amount of chosen milliseconds, Sleep() function is used only with Windows systems so to use it we must include <windos.h>

3- void usleep(int microseconds) which forces the whole program to be delayed to the amount of chosen microseconds, usleep() function is used only with Linux systems so to use it we must include <unistd.h>

important note: we wrapped Sleep() and usleep() functions in our delay() function to make it OS independent and once the machine system is detected while compiling, the appropriate function will be used

4- int String.length() which returns the number of characters of the calling string, we use it while changing the CPU time limit according to the user choice and pass by all digits of the received input from the user digit by digit to check that it's a number digit, if only one digit of the whole input was not a number digit, we print a warning message to the user asking him to enter only number digits not letters, to use this function We must include <string>

- **A user guide with snapshots**

1- A welcome message to the user:

```
E:\AI\Mancala\Debug>Mancala
Welcome to MANCALA, We hope you enjoy this game.
```

2- The user is asked to determine the game difficulty, please note that the three characters in the image below are case-insensitive, if he entered one of these 3 characters, step 3 will be skipped:

```
Please select game difficulty, type e for easy, m for medium, or h for hard.
M
You selected medium level, type c to confirm or anything else to repeat your choice.
```

3- If the user entered anything different from the above 3 characters, step 2 will be repeated:

```
Please select game difficulty, type e for easy, m for medium, or h for hard.
t
Invalid choice!! please enter only one of the 3 valid characters.
Please select game difficulty, type e for easy, m for medium, or h for hard.
```

4- Check if the user already wants the selected level or it might be a wrong character and wants to choose another difficulty level, if he wants to confirm the selected level he enters ‘c’ or ‘C’, and step 5 will be skipped:

```
You selected medium level, type c to confirm or anything else to repeat your choice.
C
```

5- If he wants to choose another difficulty level, he enters anything else and then we go back to step 2:

```
You selected medium level, type c to confirm or anything else to repeat your choice.
u
Please select game difficulty, type e for easy, m for medium, or h for hard.
```

6- The user is asked to determine the game mode, please note that the two characters in the image below are case-insensitive, if he entered one of these 2 characters, step 7 will be skipped:

```
Please select game mode, type w for with-stealing mode or n for no-stealing mode.
w
You selected with-stealing mode, type c to confirm or anything else to repeat your choice.
```

7- If the user entered anything different from the above 2 characters, step 6 will be repeated:

```
Please select game mode, type w for with-stealing mode or n for no-stealing mode.
p
Invalid choice!! please enter only one of the 2 valid characters.
Please select game mode, type w for with-stealing mode or n for no-stealing mode.
```

8- Check if the user already wants the selected mode or it might be a wrong character and wants to choose the other mode, if he wants to confirm the selected mode he enters ‘c’ or ‘C’, and step 9 will be skipped:

```
You selected with-stealing mode, type c to confirm or anything else to repeat your choice.
c
```

9- If he wants to choose the other game mode, he enters anything else and then we go back to step 6:

```
You selected with-stealing mode, type c to confirm or anything else to repeat your choice.
q
Please select game mode, type w for with-stealing mode or n for no-stealing mode.
```

10- The user is asked to determine his playing order, please note that the two characters in the image below are case-insensitive, if he entered one of these 2 characters, step 11 will be skipped:

```
Please select your turn order,  type f to be first player or s to be second player.
S
You chose to be second player, type c to confirm or anything else to repeat your choice.
```

11- If the user entered anything different from the above 2 characters, step 10 will be repeated:

```
Please select your turn order,  type f to be first player or s to be second player.
4
Invalid choice!! please enter only one of the 2 valid characters.
```

12- Check if the user already wants the selected playing order or it might be a wrong character and wants to choose the other playing order, if he wants to confirm the selected playing order he enters ‘c’ or ‘C’, and step 13 will be skipped:

```
You chose to be second player, type c to confirm or anything else to repeat your choice.
c
```

13- If he wants to choose the other playing order, he enters anything else and then we go back to step 10:

```
You chose to be second player, type c to confirm or anything else to repeat your choice.
B
Please select your turn order,  type f to be first player or s to be second player.
```

14- The user is asked if he wants to change the CPU time limit or keep it as default (5 seconds):

```
Default CPU time limit is 5 seconds, press y to change it or anything else to keep it as default
```

15- If he entered anything except ‘y’ or ‘Y’, then CPU time limit is set to 5 seconds, and we will go directly to step 19.

```
Default CPU time limit is 5 seconds, press y to change it or anything else to keep it as default
.
n
Game will begin in 5 seconds, please know that your left most pocket number is 1 and your right
most pocket is 6.
```

16- If he entered ‘y’ or ‘Y’, then he is asked to enter the CPU time limit in seconds but in the numerical form only:

```
Default CPU time limit is 5 seconds, press y to change it or anything else to keep it as default
.
y
Please write the time limit in seconds.
```

17- If he entered a number consisting of only digit(s) from 0 to 9, then the CPU time limit is set to this value and the unit is second, and step 18 will be skipped

```
Please write the time limit in seconds.
2
```

.

18- if he enters any non-numerical digit, then we go back to step 17:

```
Please write the time limit in seconds.
two
Please do not write any letter, just write a number .
Please write the time limit in seconds.
```

19- A message will inform the user that the game will begin in 5 seconds, the lower pockets are the user pockets while the upper pockets are CPU pockets, the pockets of the user are numbered from 1 to 6 starting from the left:

```
Game will begin in 5 seconds, please know that your left most pocket number is 1 and your right most pocket is 6.
```

20- The initial state of the board will be displayed to the user, and it will be displayed later after any play from either the user or the CPU until the game ends:

```
 _____________________________
|   |4|4|4|4|4|4|
|0-----------------0|
|   |4|4|4|4|4|4|
 _____________________________
```

21- In any of the user turns, he is asked to select a non-empty pocket: the selection requires entering a number from 1 to 6 not anything else or even its string representation, if he enters a valid number of non-empty pocket, steps 22 and 23 will be skipped:

```
CPU turn, please wait
 _____________________
|  |2|6|8|0|0|0|
|11-------------9|
|  |1|0|0|8|1|2|
 _____________________

your turn, please select one of non-empty pockets of you
1


 _____________________
|  |2|0|8|0|0|0|
|11-------------16|
|  |0|0|0|8|1|2|
 _____________________
```

22- If he entered a number of an empty pocket, then we will go back to step 21:

```
CPU turn, please wait
 ____________________
|  |2|6|7|1|0|0|
|11-------------9|
|  |1|0|3|7|0|1|
 ____________________

your turn, please select one of non-empty pockets of you
2
invalid choice, use only valid numbers of non-empty pockets
your turn, please select one of non-empty pockets of you
```

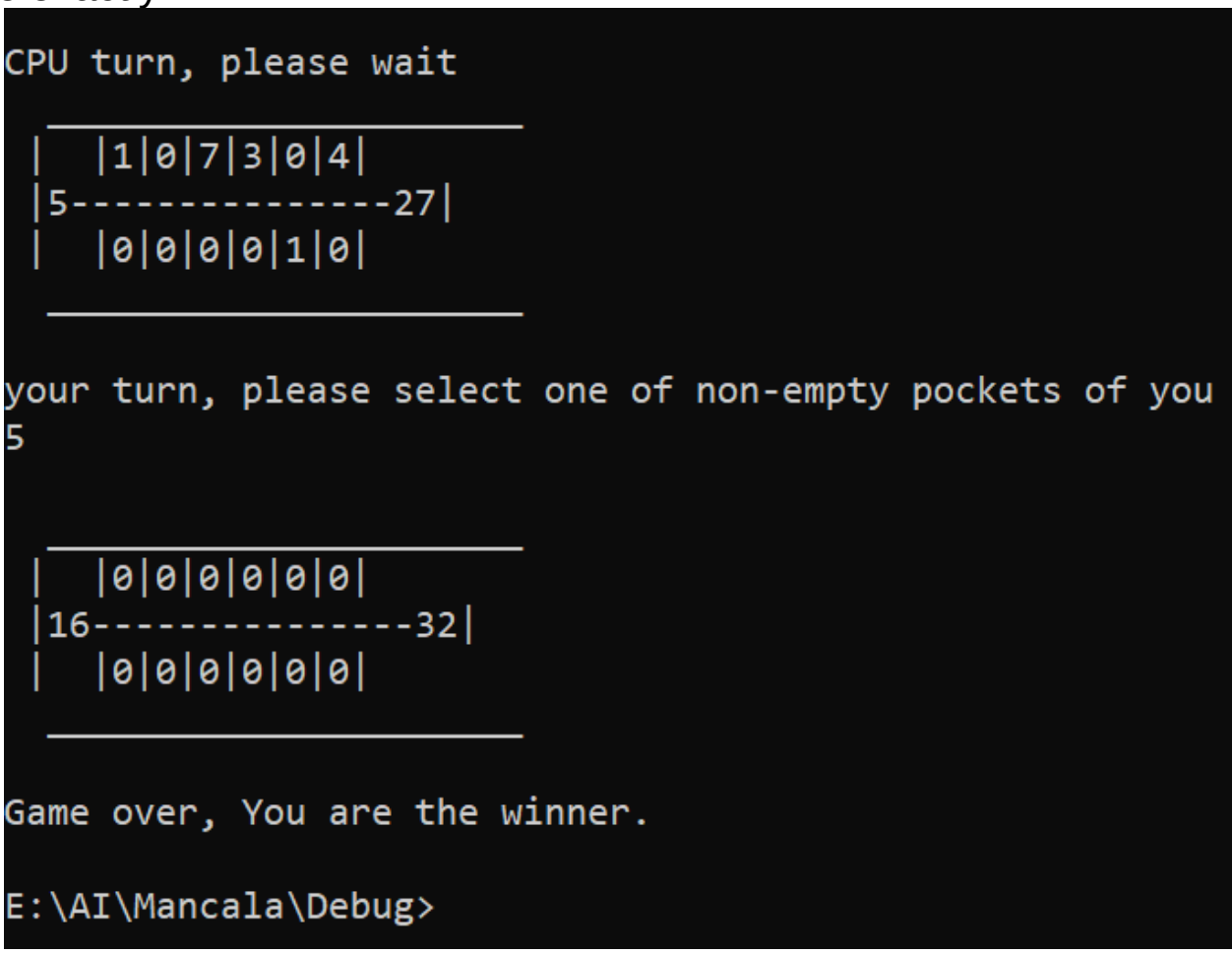
23- If he entered a string or a number outside the range (1:6) then we go back to step 21:

```
your turn, please select one of non-empty pockets of you
7
invalid choice, use only valid numbers from 1 to 6 and not strings
```

24- If the game did not end, and the last ball of any player at any of his turns was thrown to his mancala, he will be given another try and in case of human, we will go back to step 21:

```
CPU turn, please wait
  _____________________
 |  |0|1|9|1|2|0|
 |14---------------18|
 |  |1|0|0|0|1|1|
  _____________________

your turn, please select one of non-empty pockets of you
6


  _____________________
 |  |0|1|9|1|2|0|
 |14---------------19|
 |  |1|0|0|0|1|0|
  _____________________

your turn, please select one of non-empty pockets of you
1


  _____________________
 |  |0|0|9|1|2|0|
 |14---------------21|
 |  |0|0|0|0|1|0|
  _____________________
```

25- When the game ends, the last state of the board will be displayed and the player whose Mancala balls is greater than 24 is the winner, or it's draw if both Mancalas balls are exactly 24:



- **Who did what**

First, all of us participated in the design phase through 3 zoom meetings in 3 consecutive days then we split the implementation phase as following:

1- Muhammed will be responsible of the implementation of the receive_user_choice() function which receives the user selected mode(stealing or no-stealing), the user selected turn(first or second), the user selected level (easy, medium, or hard), and finally the user selected timeout in case he wants to change it and didn't use the default.

2- Ahmed will be responsible of the implementation of display_Mancala() function which receives the current state of the whole board and display it showing the upper half as the CPU part and the lower half as the HUMAN part and all the moves are counterclockwise, he will be responsible of the implementation of the delay() function and we asked him to make this function OS independent, and finally he will be responsible of the implementation of get_score() function which is the only metric to consider this state as a good state or bad state with respect to CPU

3- Fares will be responsible of the implementation of is_valid() function which take the current state and the next move and check if this move is valid or not depending on if this pocket contains balls or empty, he will also be responsible of the implementation of the get_next_state_indices() function which returns the indices of all children of the current state including valid and invalid children and it shows the difference between them .

4- Omar will be responsible of the implementation of minimax() function which apply the minimax algorithm with alpha-beta pruning with a variable depth depending on the game difficulty level which is selected by the user, he will also be responsible of integrating all the functions in a

single source file with a single header file to make it easier for building, reading, and maintenance.

5- Amir will be responsible of the implementation of get_next_state() function which takes the current state and the played move and returns the next state which may be different when it's stealing or no-stealing mode indicating if there's another turn or if the game ended, he will also be responsible of using all of the previous functions in the main and integrating them efficiently