|  |  |
| --- | --- |
| **Project Case** | Description: LogoBINUS-University |
| Data Structure |
| **Periode Berlaku** Semester Genap 2020/2021  ***Valid on*** *Even Year 2020/2021* | **Software Laboratory Center**  **Assistant Recruitment 21-2** |

## Soal

*Case*

**cABsa**

Your friend, Arthur Curry is fond of the card game Capsa, you can read more about it here : (https://medium.com/@alonabaru/bermain-capsa-susun-untuk-pemula-e46a9c008e1a). **Capsa is a game of playing cards with 4 players, and each player has 13 cards in their hands** hence the word Cap-sa / 十三 / Thirteen in Hokkien dialect. Mr. Arthur Curry is in quarantine alone, without internet, and only gets delivery of food and necessity once a week. You as his good friend want him to be happy in his quarantine, so **you plan to create a capsa computer game that can run on any pc** and you will send it to him**.** You are learning about data structure in C so you decided to **use data structure as a base for your program**. **You don’t have to create the capsa game exactly like the real-world capsa game**, you just have to follow the following requirements :

* **Prerequisite**
  + Prepare the cards to be dealt
    1. There are **52 cards on the deck, each card has its own unique identification number** starting from the lowest (3 diamond) to the highest (2 spade). E.g. : 3 diamond’s id is one, 3 club’s id is two, 3 heart’s id is three, 3 spade’s id is four, 4 diamond’s id is five, and so on.
    2. Shuffle the deck using **Fisher-Yates shuffle algorithm**
    3. **Deal the cards to the 4 players equally** so that each player have 13 cards on his hands
    4. **Each combo card has its own id** by adding the id of each card that builds the combo.
  + **Use Breadth-First Search to find the best card to be played** (the lowest card or combo which have value that is above the last played)
  + Please **use your creativity in designing the struct and the gameplay**.
  + The **system must be able to analyze the combo** such as pair, three of a kind, and full combos of each player.
  + You **only need to show the player’s cards and the last card played**. **You are not required to design the interface beautifully.**
  + For the design purposes in the example program, cards with the number 10 are shown as X.
* **Start Game**
* **Prompt** userto **input his player number**. **Validate** the input must **between 1 and 4 inclusively**.

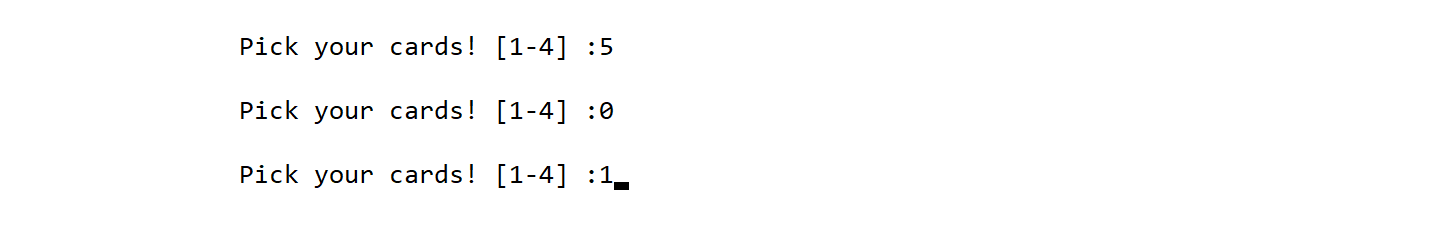


Figure 1. Pick cards.

* Assign the user as the player with the number of his choosing.
* **Gameplay**
  + The **player who has the three diamond will start the game**. Show the player number!

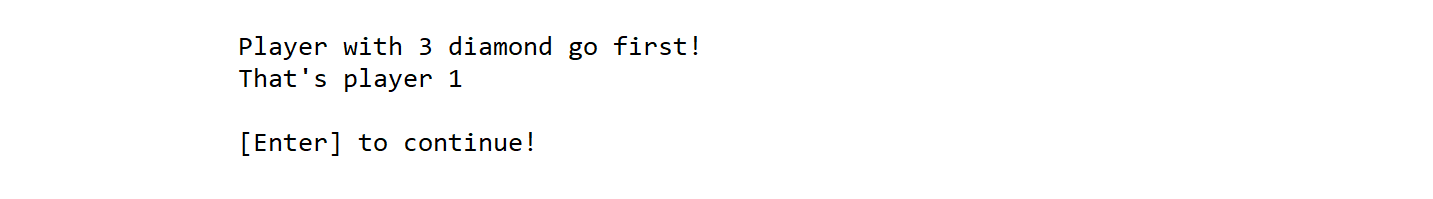
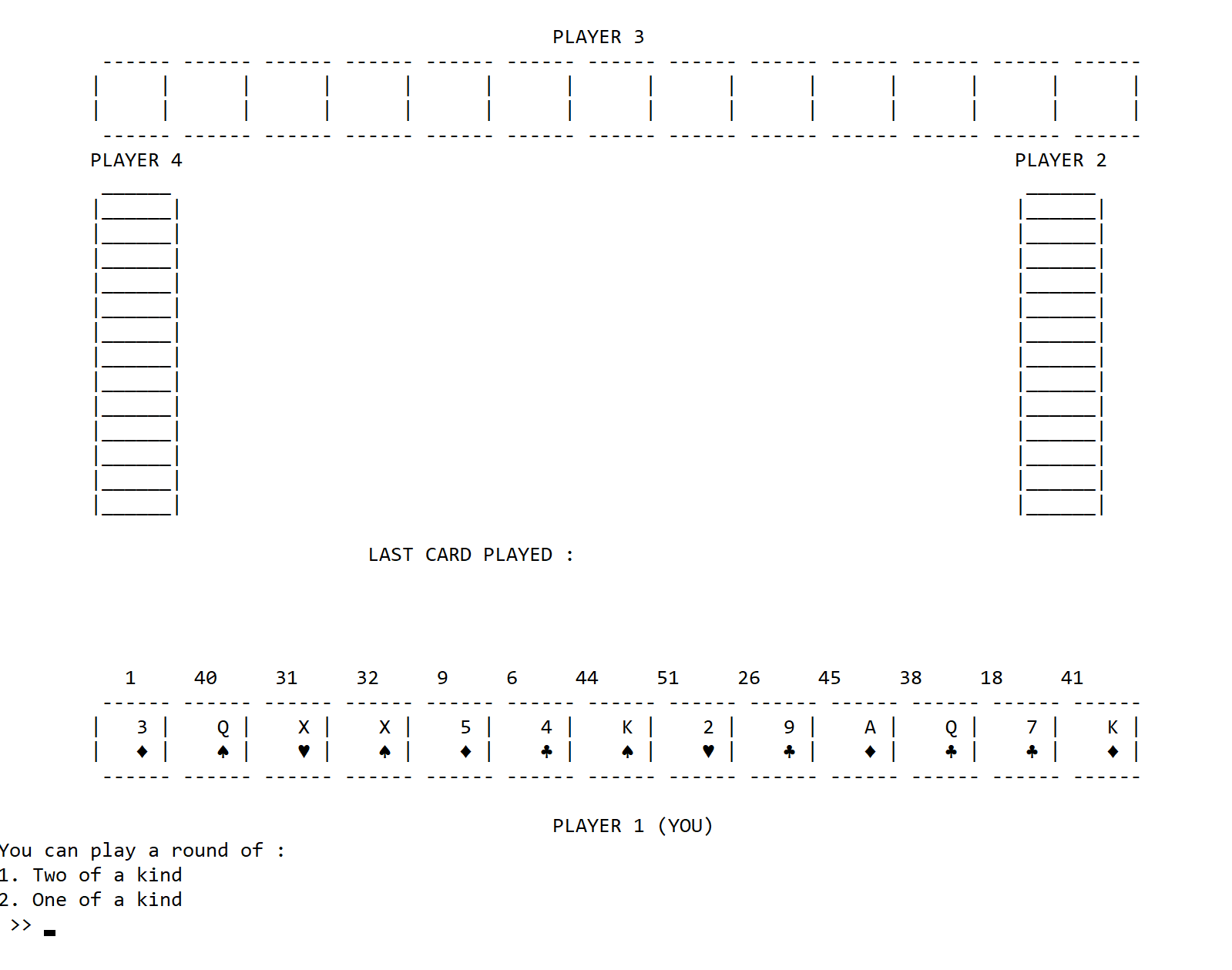


Figure 2. The player with three diamond.

* + If the player who will start the **new round is a computer**:
    1. If the player **has a full combo** in his cards then **find the lowest full combo value** and discard it.
    2. If the player **does not have a full combo** then **find the lowest three of a kind combo value** and discard it.
    3. If the player **does not a three of a kind combo**, then **find the lowest pair combo** and discard it.
    4. If the player **does not have a pair combo** then **discard the single lowest card value**.
  + If the player who will start the **new round is the user**:
    1. **Show a menu to let the user choose what kind of combo the user wants to play** (only show combo(s) that he owns)

 Figure 3. Available combos in hands

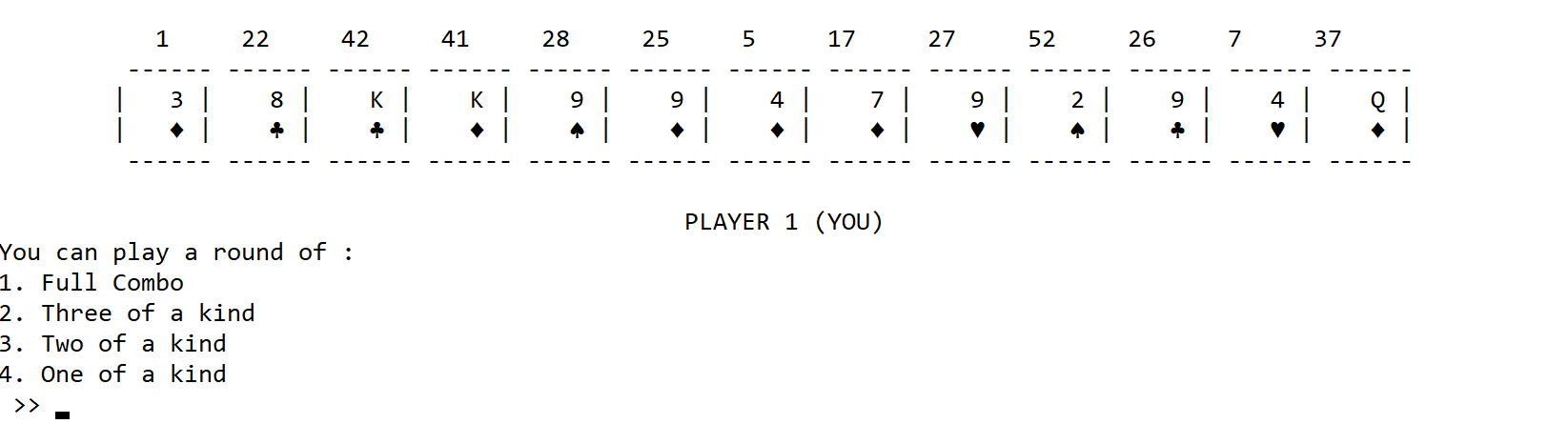
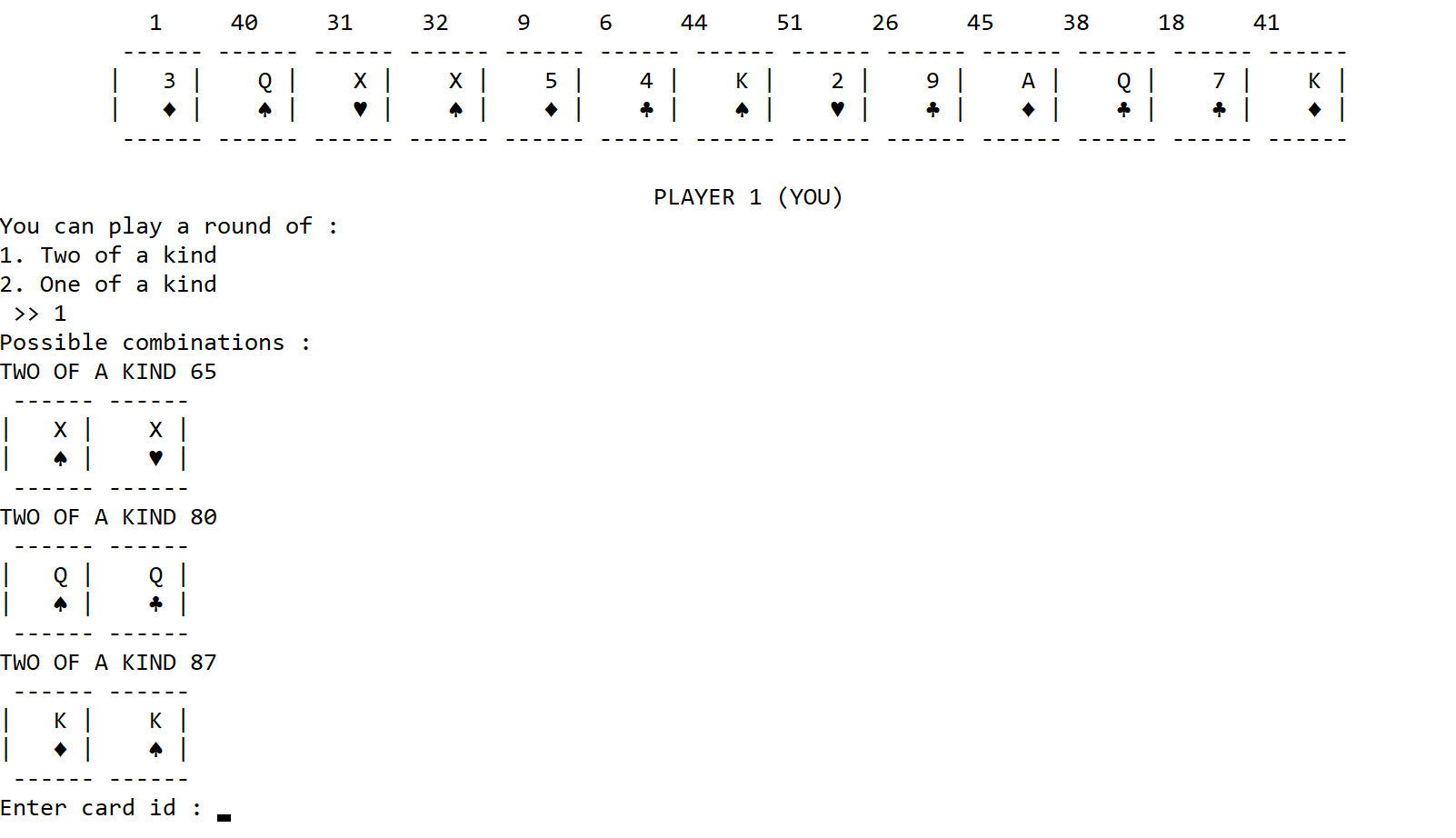
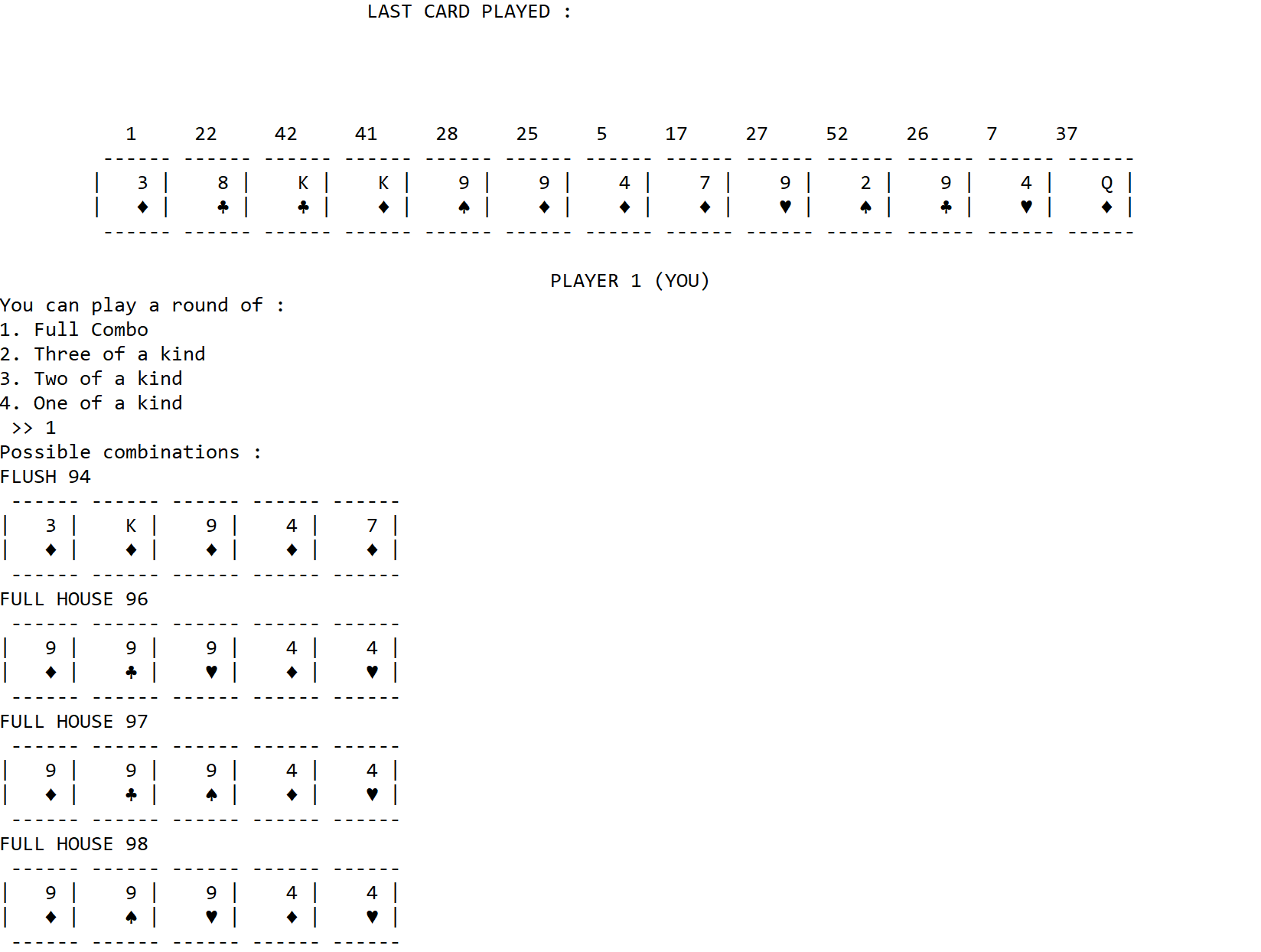


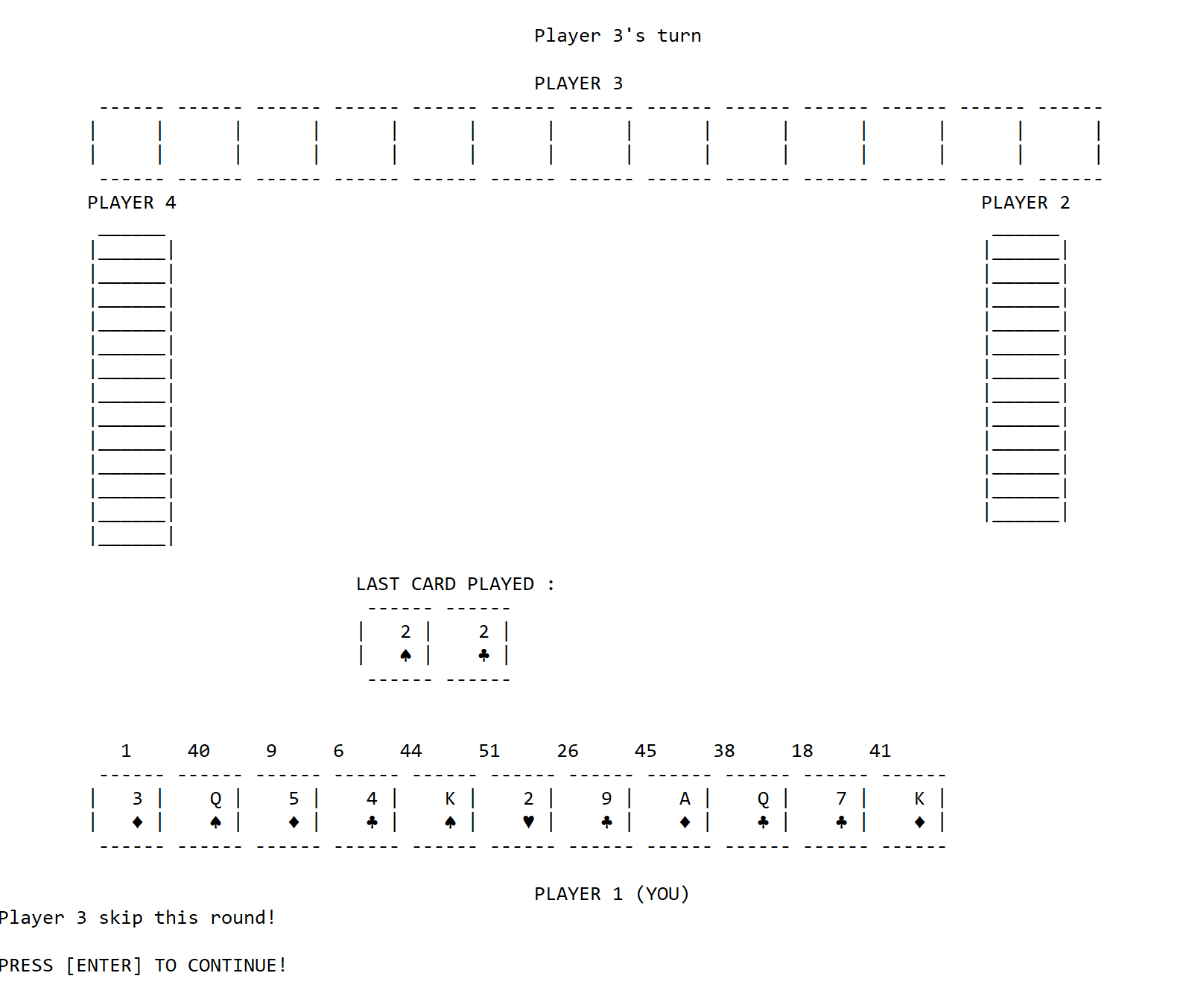
Figure 4. Available combos in hands.

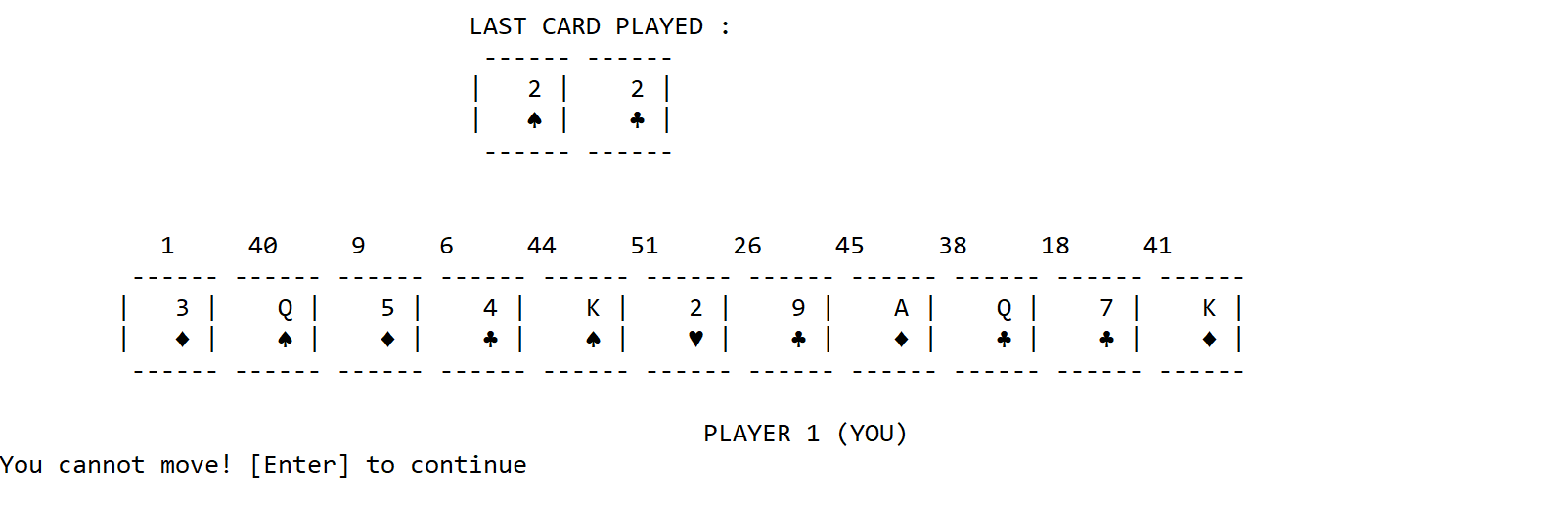
* + 1. **If the user chose anything except the one of a kind menu**, show the available combo from the chosen menu.

 Figure 5. The user chooses to play two of a kind game and shown the available two of a kind combos in his hands.

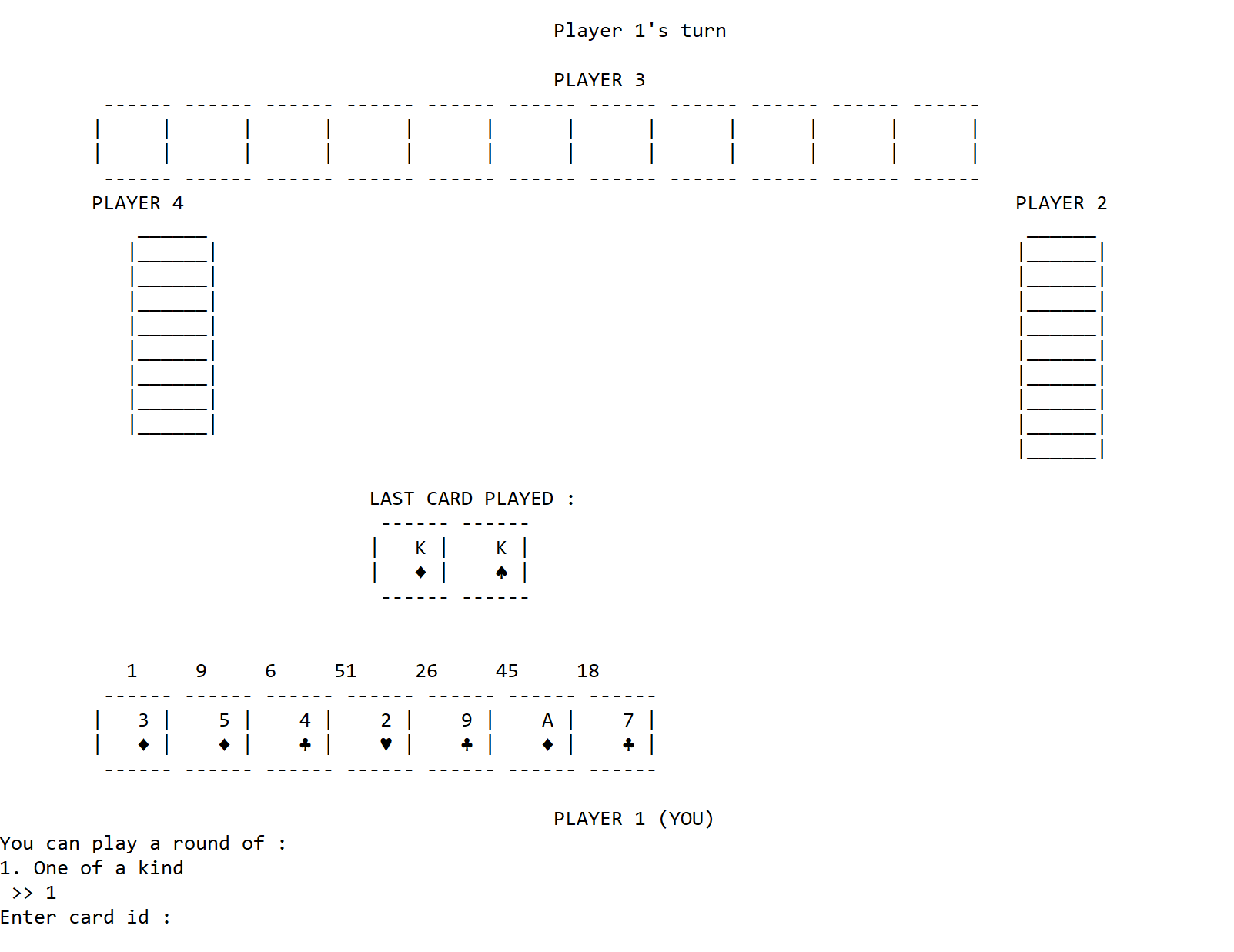
 Figure 6. The User chooses to play full combo and shown the available full combos with their ids.

* + - * Ask the user to **input the id card combo**
      * Validate whether the combo with the inputted id exists.
      * **Discard the chosen** **combo** if it exists.
    1. If the user chose to play one of a kind game:
       - Ask user to input the card id as shown at the top of his card
       - Validate whether the card with the inputted id exists.
       - **Discard the chosen** **card** if it exists.
  + Then the **gameplay will continue just like a *capsa* game**.
  + **If a player skips his turn in this round then shows that he skips this round**, **he may not play again** until the current round is finished and a new round is started.

 Figure 7. Player 3 skipped this round.

Figure 8. Player 1 (user) cannot move so he skipped this round automatically.

* + If no more players have a higher card value or higher combo value, then **the last player who discarded the card(s) will be the player who opens a new round**.

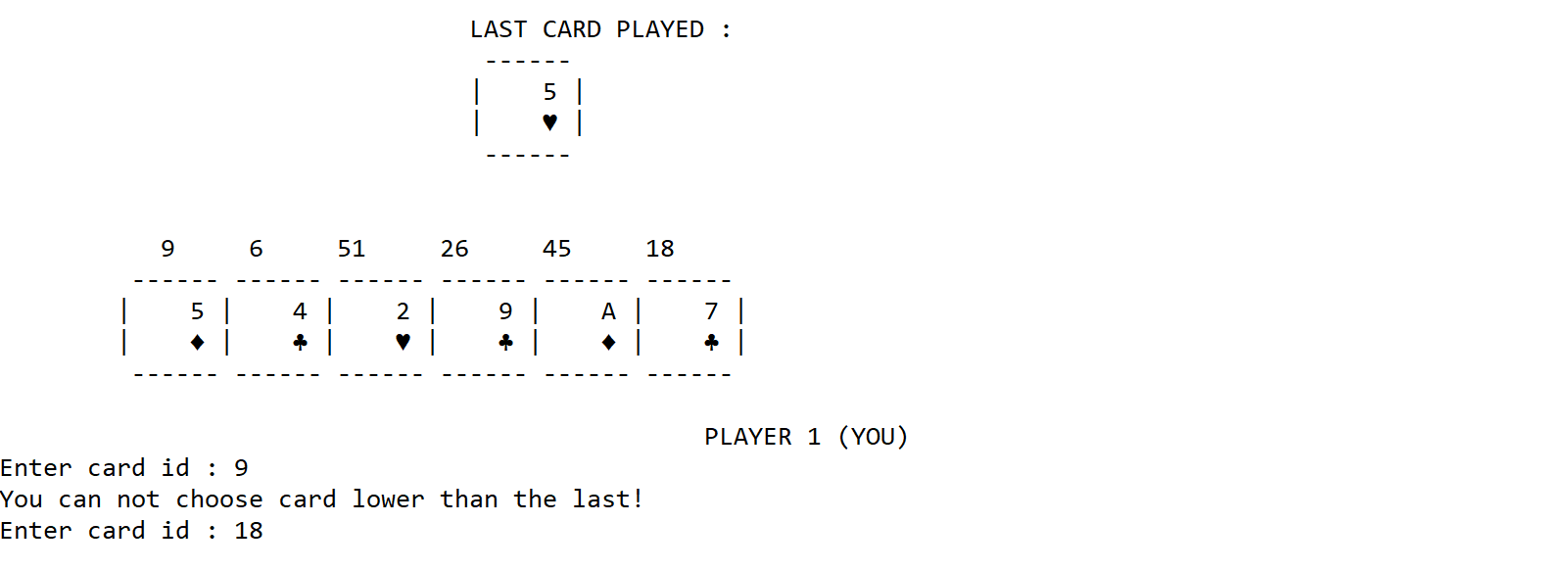
 Figure 9. Player 1 (user) discarded the pair of kings and no one can move anymore so he starts a new round.

* + Here are some rules that will apply in this *capsa* game version of yours:
    1. **If the played card on the round opener is just a single card**, then the current player must discard a single card that has a higher value than the previous.
       - The order of weight for each symbol is as the following:

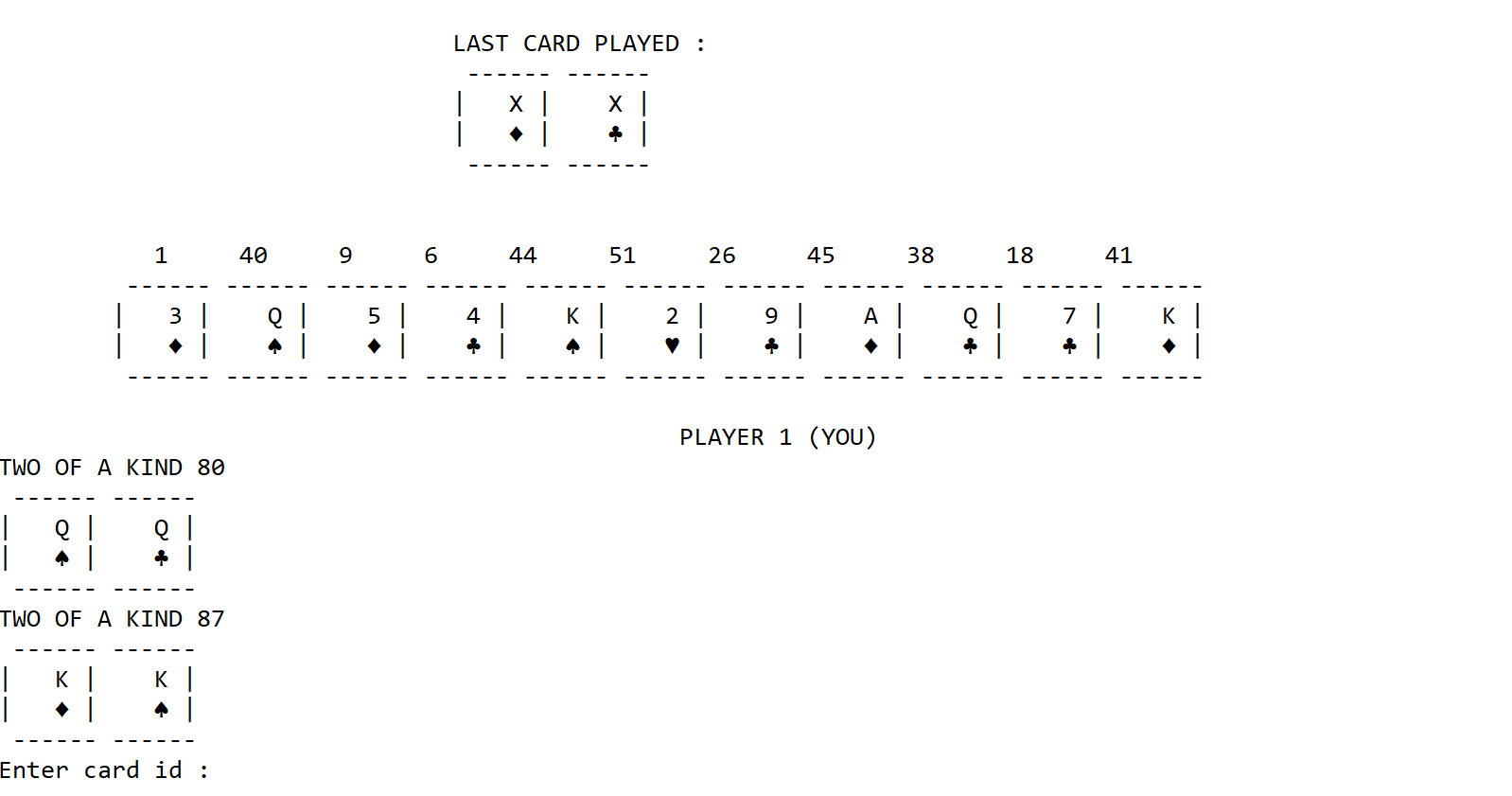
♦<♣<♥<♠

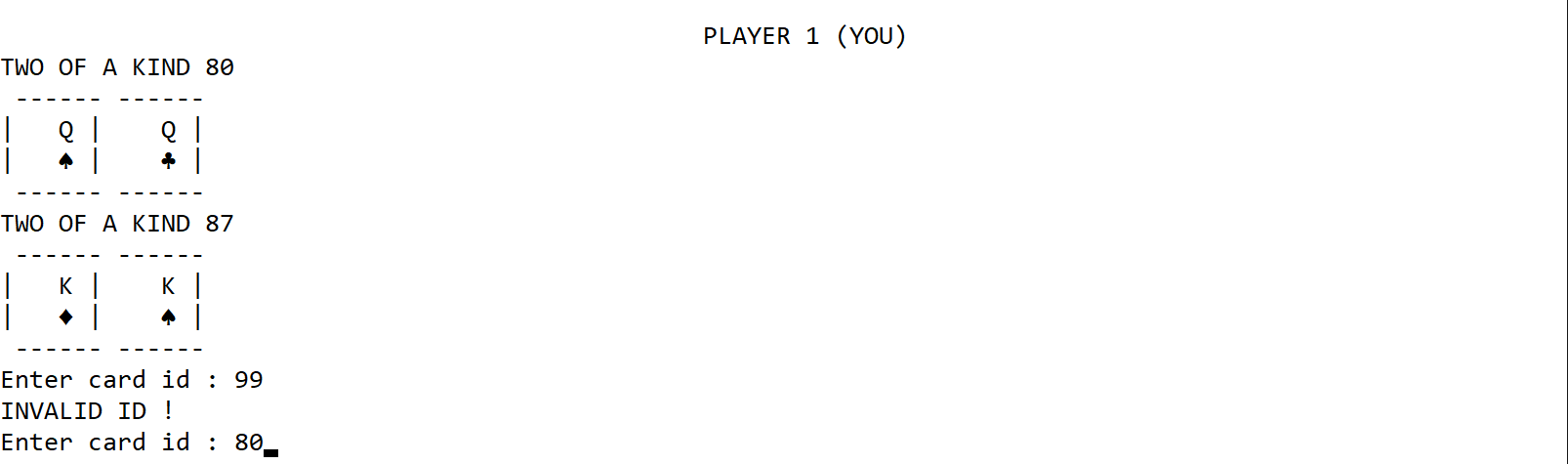
Figure 10. Symbol hierarchy.

* For example: if the previous card is 3 diamond, and the current player has 3 club then the current player **may discard the 3 club**. Or **he may discard a card with a higher number.**

 Figure 11. The last played card is 5 of hearts and the user wanted to discard 5 of diamonds.

* If the current player is a computer, **check if his weakest card which value is above the previous card, and if that card belongs to a group of combos** (Full house, Straight, etc.) **then the current player will skip his turn**. Otherwise, the **computer will discard that card**.
* If the current player is the user, o**nly allow the user to discard a single card and the value must be higher than the previously played card**. The player will discard a card by inputting the id number shown above the cards.
  + 1. **If the played card on the round opener is a pair combo**, the rules are the same as above except that if it belongs to a combo, it will still be played.
       - If the current player has no pair combo, the program will **automatically skip his turn.**
       - If the current player is the user, if he does have pair combo(s) then **show all the available pair combo that can be played and let the user choose cards combo to be discarded.**

 Figure 12. Show available pair(two of a kind) that can be played.

 Figure 13. User entered invalid combo id.

* + - * If the current player is a computer then **discard his weakest pair combo that still has a higher value compared to the last card played**.
    1. **If the played card on the round opener is a three of a kind** combo, the rules are the same as above except that if it belongs to a combo, it will still be played.
       - If the current player has no three of a kind combo, **the program will automatically skip his turn.**
       - If the current player is the user, if he does have three of a kind combo(s) then **show all the available three of a kind combo that can be played and let the user choose a card combo to be discarded**.
       - If the current player is a computer then **discard his weakest three of a kind combo that still has a higher value compared to the last card played.**
    2. **For the five cards combo as the round opener** here is how the player will play in the round.
       - In this version of *capsa*, **you do not have to follow the order of weight for each combo** (e.g. normally Full House is above Flush but you may ignore this rule in this game). You just have to **make sure that the sum of each value in the combo is higher than the last.**
       - If the current player has no five card combo (full combo), the program will **automatically skip his turn.**
       - If the current player is the user, if he does have full combo(s) then **show all the available full combo(s)** that can be played and **let the user choose cards combo to be discarded.**
       - If the current player is a computer then **discard his weakest three of a kind combo that still has a higher value compared to the last card played**.
* **Winner**
  + **Display the winning player number**, show your favorite quote, and end the game.

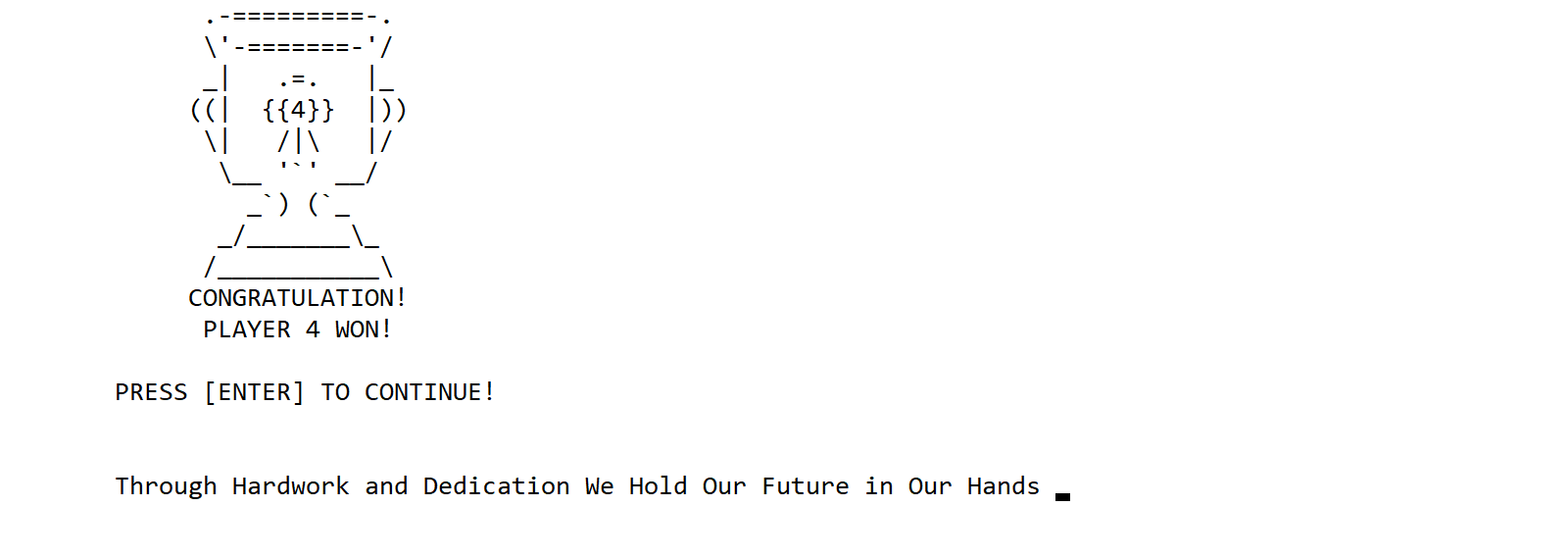


Figure 14. Show winner screen.

Please run the EXE file to see the sample program.