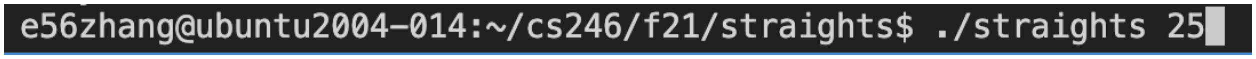
Demo

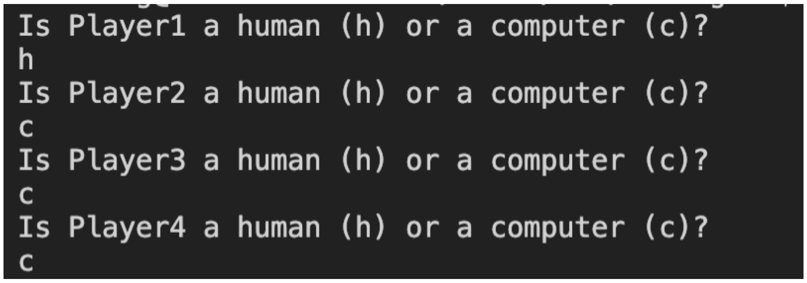
<u>Testing human commands (test1.in):</u>

User runs the program with seed 25



```
e56zhang@ubuntu2004-014:~/cs246/f21/straights$ ./straights 25
```

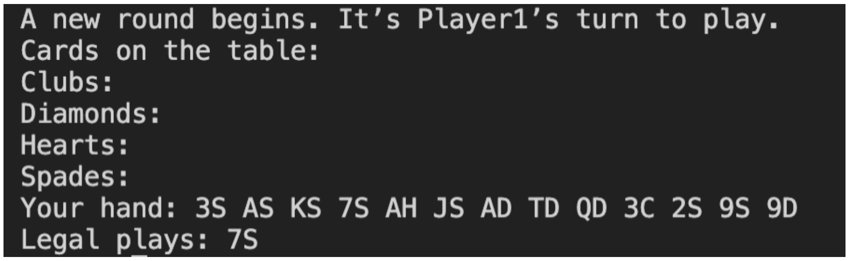
User enters these characters individually: { h, c, c, c } (pressing enter in between them)



```
Is Player1 a human (h) or a computer (c)?
h
Is Player2 a human (h) or a computer (c)?
c
Is Player3 a human (h) or a computer (c)?
c
Is Player4 a human (h) or a computer (c)?
c
```

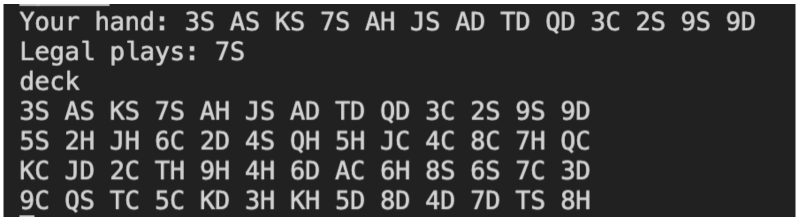
User is given:

- who the first player is (player 1 because they hold 7S)
- Cards on the table (empty because no cards have been played)
- Your hand
- Legal plays (since this is the first turn, the only legal play available is 7S)



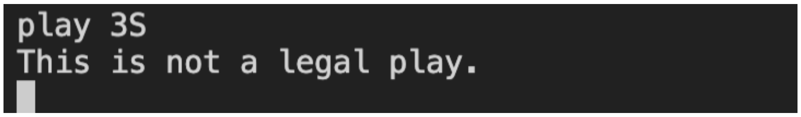
```
A new round begins. It’s Player1’s turn to play.
Cards on the table:
Clubs:
Diamonds:
Hearts:
Spades:
Your hand: 3S AS KS 7S AH JS AD TD QD 3C 2S 9S 9D
Legal plays: 7S
```

User types deck and sees the deck (note that the first 13 cards correspond to player 1’s hand)



```
Your hand: 3S AS KS 7S AH JS AD TD QD 3C 2S 9S 9D
Legal plays: 7S
deck
3S AS KS 7S AH JS AD TD QD 3C 2S 9S 9D
5S 2H JH 6C 2D 4S QH 5H JC 4C 8C 7H QC
KC JD 2C TH 9H 4H 6D AC 6H 8S 6S 7C 3D
9C QS TC 5C KD 3H KH 5D 8D 4D 7D TS 8H
```

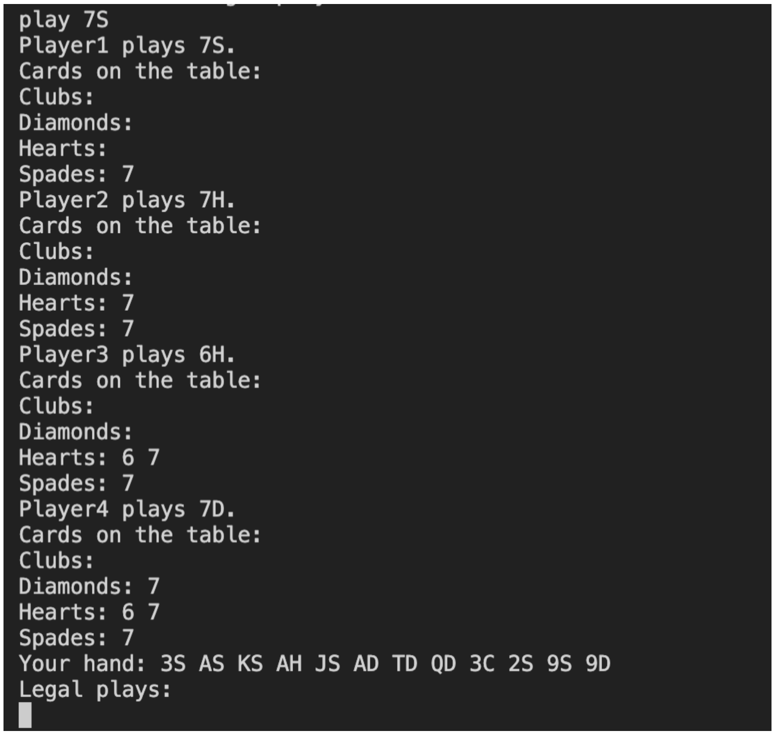
User plays 3S (in user’s hand) and is met with an error message because player 1 must play 7S and the chance to re-enter another card



```
play 3S
This is not a legal play.
```

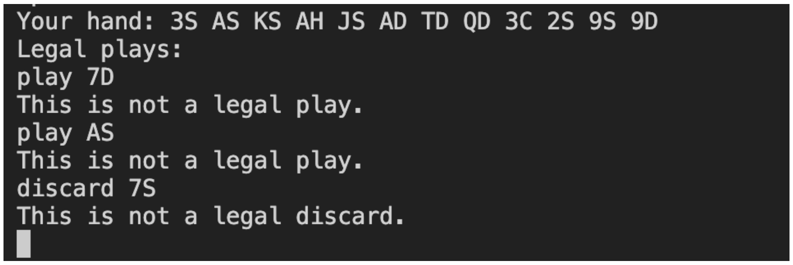
User plays 7S and since we are playing with cpus, this skips to user’s next turn
As you can see:

- The 7S that player 1 played is on the table
- Player 2 plays 7H because that's their first legal play (refer to deck)
- Player 3 plays 6H because that's their first legal play (refer to deck)
- Player 4 plays 7D because that's their first legal play (refer to deck)
- When it comes back to player 1’s turn 7S is no longer in their hand



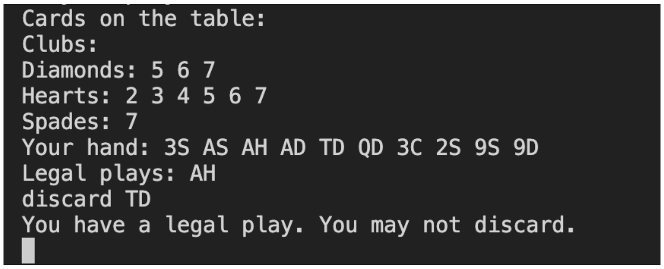
```
play 7S
Player1 plays 7S.
Cards on the table:
Clubs:
Diamonds:
Hearts:
Spades: 7
Player2 plays 7H.
Cards on the table:
Clubs:
Diamonds:
Hearts: 7
Spades: 7
Player3 plays 6H.
Cards on the table:
Clubs:
Diamonds:
Hearts: 6 7
Spades: 7
Player4 plays 7D.
Cards on the table:
Clubs:
Diamonds: 7
Hearts: 6 7
Spades: 7
Your hand: 3S AS KS AH JS AD TD QD 3C 2S 9S 9D
Legal plays:
```

User has no legal play but tries playing 7D but that is not even in the user's hand, the user then tries to play AS which is in the user's hand but is not a legal play. User then tries to discard 7S which is invalid. All of these produce an error and prompt a re-input



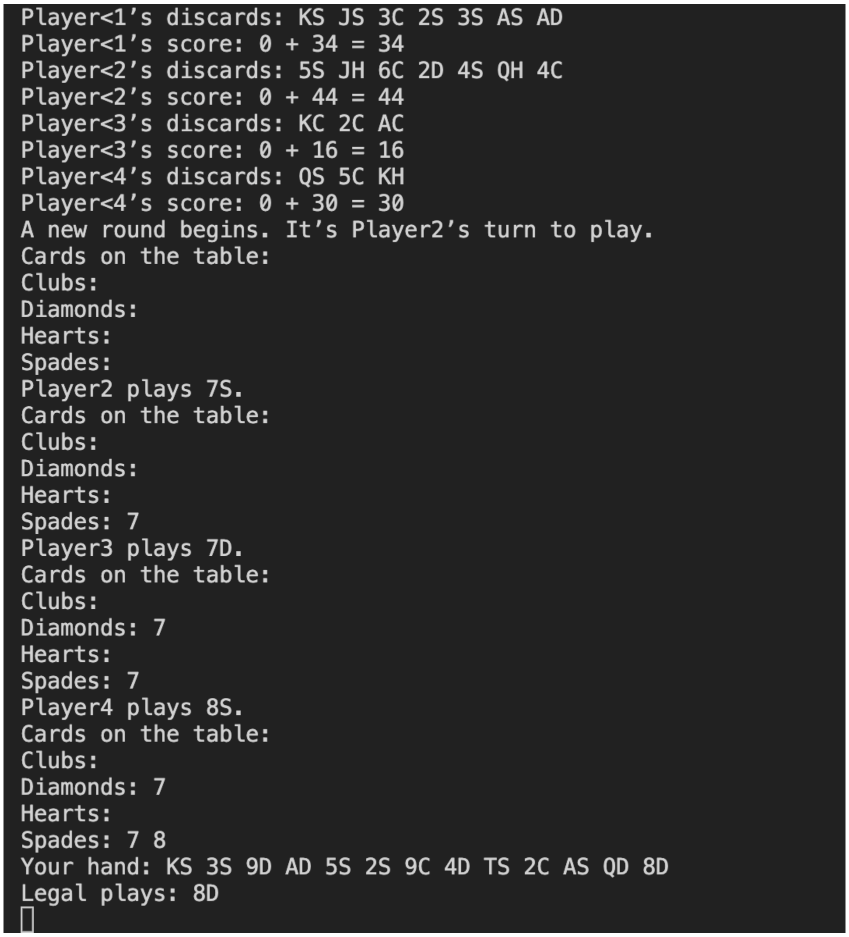
```
Your hand: 3S AS KS AH JS AD TD QD 3C 2S 9S 9D
Legal plays:
play 7D
This is not a legal play.
play AS
This is not a legal play.
discard 7S
This is not a legal discard.
```

User discards KS, and discards JS on the turn afterwards. Their hand now no longer contains either of those cards because it's in the discard pile. User then tries to discard TD but cannot because they have a legal play



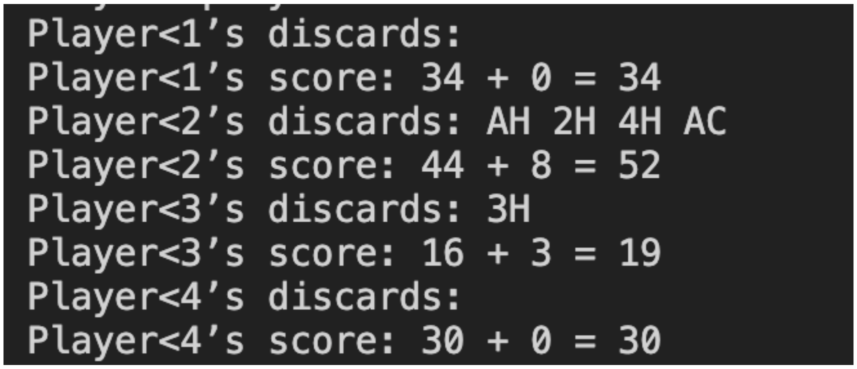
```
Cards on the table:
Clubs:
Diamonds: 5 6 7
Hearts: 2 3 4 5 6 7
Spades: 7
Your hand: 3S AS AH AD TD QD 3C 2S 9S 9D
Legal plays: AH
discard TD
You have a legal play. You may not discard.
```

User plays AH, 9S, 9D, TD, and then QD. On the subsequent turn there are no legal plays so user discards 3C, 2S, 3S, AS, and AD. The user is then show all the players' discards and score. For user, they discarded KS(13) + JS(11) + 3C(3) + 2S(2) + 3S(3) + AS(1) + AD(1) = 34



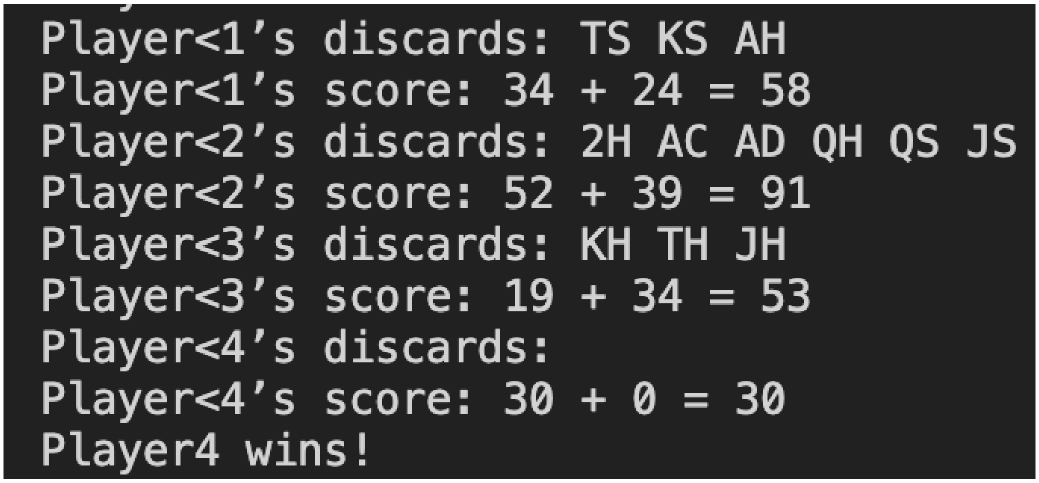
```
Player<1's discards: KS JS 3C 2S 3S AS AD
Player<1's score: 0 + 34 = 34
Player<2's discards: 5S JH 6C 2D 4S QH 4C
Player<2's score: 0 + 44 = 44
Player<3's discards: KC 2C AC
Player<3's score: 0 + 16 = 16
Player<4's discards: QS 5C KH
Player<4's score: 0 + 30 = 30
A new round begins. It's Player2's turn to play.
Cards on the table:
Clubs:
Diamonds:
Hearts:
Spades:
Player2 plays 7S.
Cards on the table:
Clubs:
Diamonds:
Hearts:
Spades: 7
Player3 plays 7D.
Cards on the table:
Clubs:
Diamonds: 7
Hearts:
Spades: 7
Player4 plays 8S.
Cards on the table:
Clubs:
Diamonds: 7
Hearts:
Spades: 7 8
Your hand: KS 3S 9D AD 5S 2S 9C 4D TS 2C AS QD 8D
Legal plays: 8D
```

User plays 8D, 9D, TS, QD, 9C, 5S, 3S, KS, 2S, 4D, AS, AD, 2C



```
Player<1's discards:
Player<1's score: 34 + 0 = 34
Player<2's discards: AH 2H 4H AC
Player<2's score: 44 + 8 = 52
Player<3's discards: 3H
Player<3's score: 16 + 3 = 19
Player<4's discards:
Player<4's score: 30 + 0 = 30
```

Since user didn't ever discard, their score never increased
User discards TS, plays 9C, 8D, 9D, 6H, 5C, 5D, 3C, JD, 3S, AS, discards KS, AH

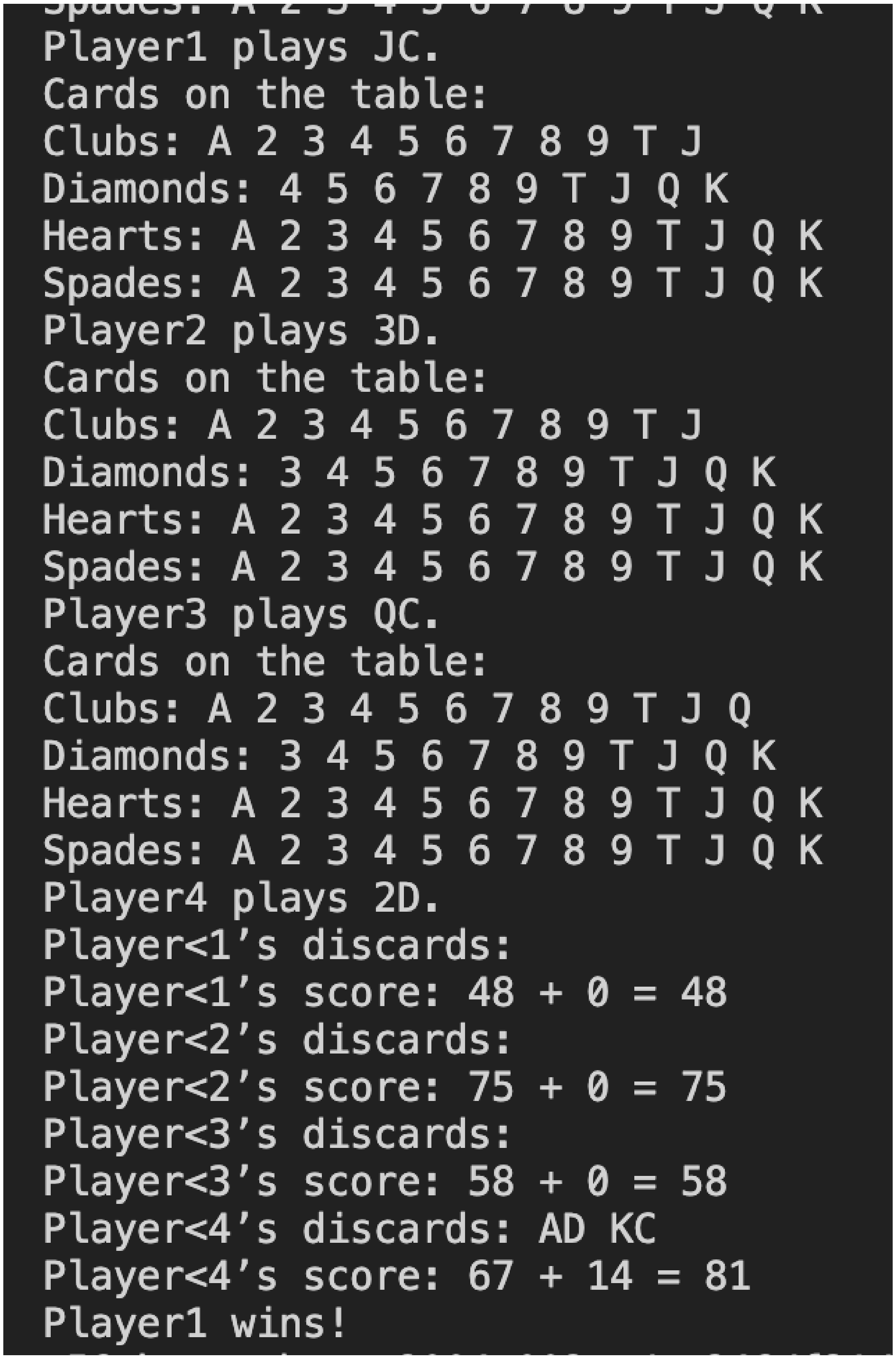


```
Player<1’s discards: TS KS AH
Player<1’s score: 34 + 24 = 58
Player<2’s discards: 2H AC AD QH QS JS
Player<2’s score: 52 + 39 = 91
Player<3’s discards: KH TH JH
Player<3’s score: 19 + 34 = 53
Player<4’s discards:
Player<4’s score: 30 + 0 = 30
Player4 wins!
```

User’s score is now 58 because of the 34 points from the previous rounds and our discards of TS + KS + AH = 24. Also the game ends because player 2 reached 80 and player 4 wins.

<u>Testing when game ends functionality and all CPU games (test2.in)</u>

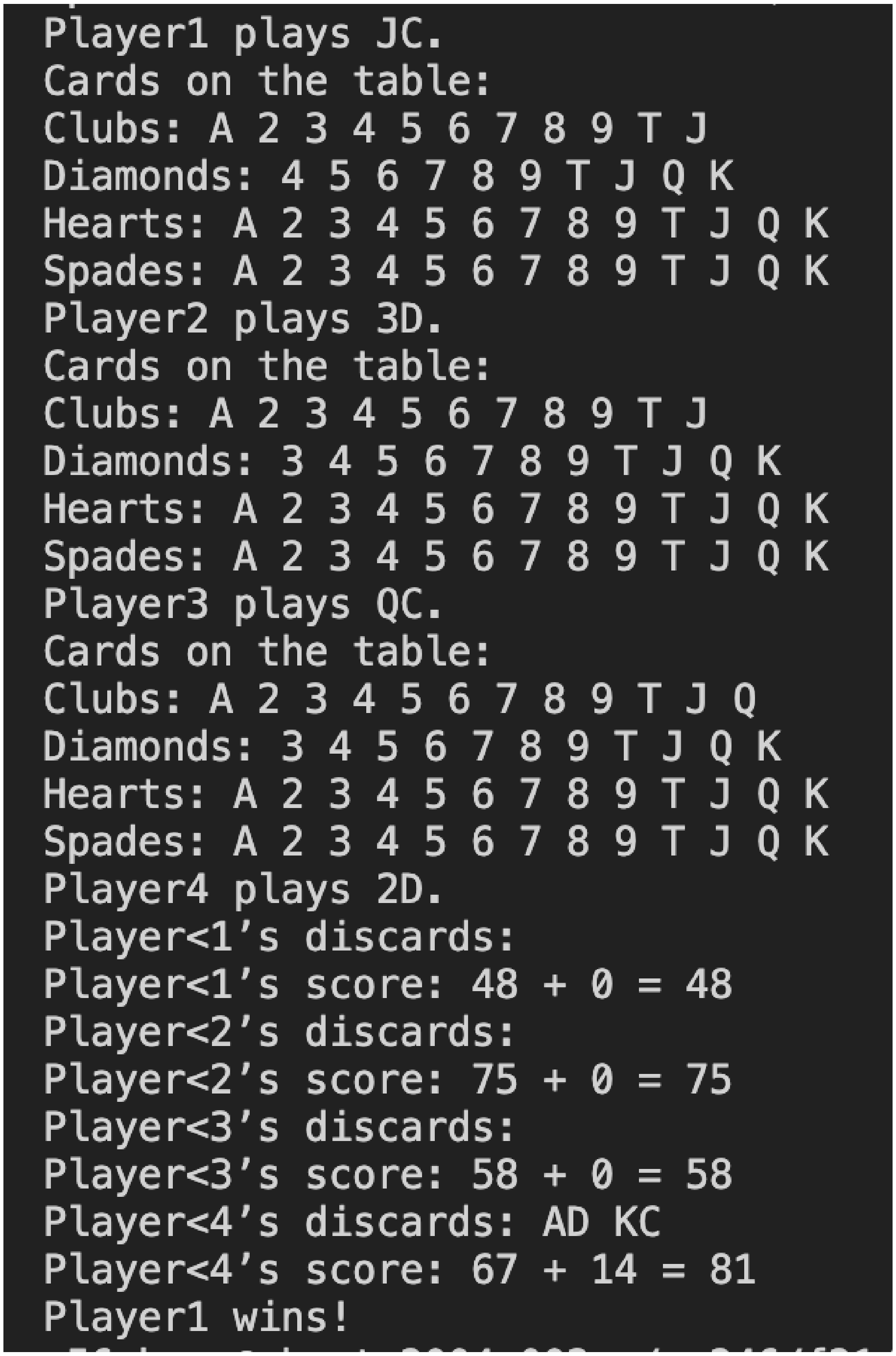
User reruns ./straights 25 but this time with all cpus, as the user can see, the game ends when one player hits 81 points and the player with the lowest score wins.



```
Spades: A 2 3 4 5 6 7 8 9 T J Q K
Player1 plays JC.
Cards on the table:
Clubs: A 2 3 4 5 6 7 8 9 T J
Diamonds: 4 5 6 7 8 9 T J Q K
Hearts: A 2 3 4 5 6 7 8 9 T J Q K
Spades: A 2 3 4 5 6 7 8 9 T J Q K
Player2 plays 3D.
Cards on the table:
Clubs: A 2 3 4 5 6 7 8 9 T J
Diamonds: 3 4 5 6 7 8 9 T J Q K
Hearts: A 2 3 4 5 6 7 8 9 T J Q K
Spades: A 2 3 4 5 6 7 8 9 T J Q K
Player3 plays QC.
Cards on the table:
Clubs: A 2 3 4 5 6 7 8 9 T J Q
Diamonds: 3 4 5 6 7 8 9 T J Q K
Hearts: A 2 3 4 5 6 7 8 9 T J Q K
Spades: A 2 3 4 5 6 7 8 9 T J Q K
Player4 plays 2D.
Player<1's discards:
Player<1's score: 48 + 0 = 48
Player<2's discards:
Player<2's score: 75 + 0 = 75
Player<3's discards:
Player<3's score: 58 + 0 = 58
Player<4's discards: AD KC
Player<4's score: 67 + 14 = 81
Player1 wins!
```

Testing ragequit (test3.in)

User reruns ./straights 25 with all human players and puts in ragequit for each player



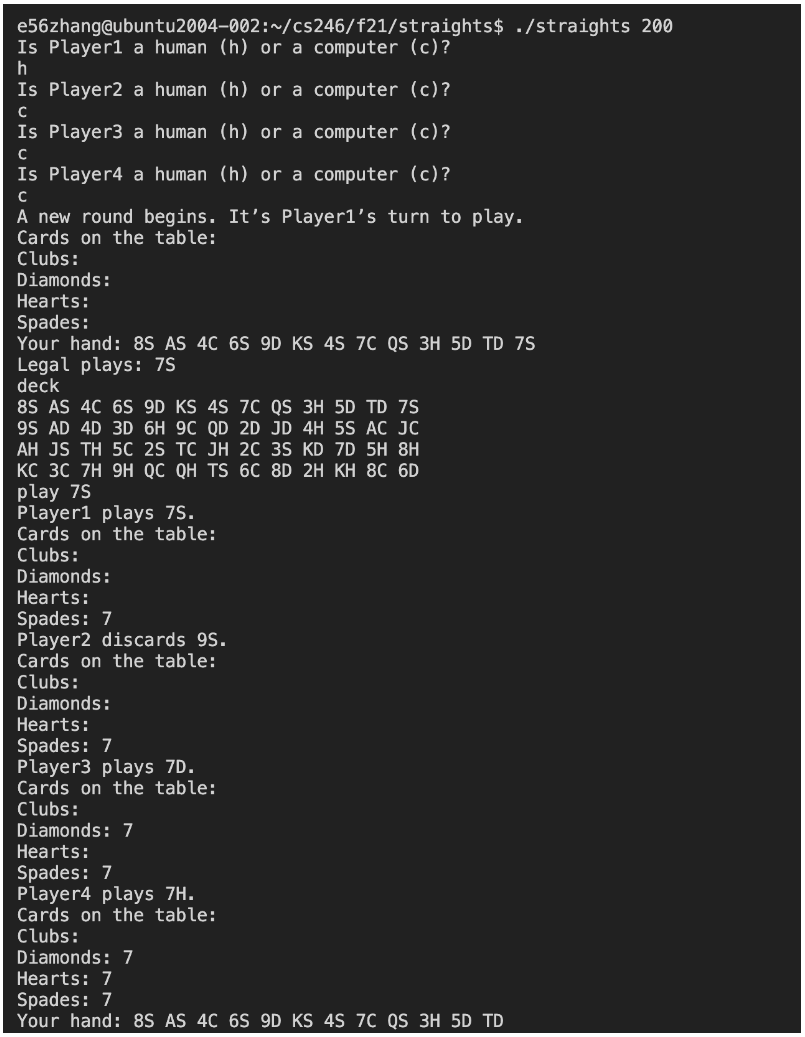
```
Player1 plays JC.
Cards on the table:
Clubs: A 2 3 4 5 6 7 8 9 T J
Diamonds: 4 5 6 7 8 9 T J Q K
Hearts: A 2 3 4 5 6 7 8 9 T J Q K
Spades: A 2 3 4 5 6 7 8 9 T J Q K
Player2 plays 3D.
Cards on the table:
Clubs: A 2 3 4 5 6 7 8 9 T J
Diamonds: 3 4 5 6 7 8 9 T J Q K
Hearts: A 2 3 4 5 6 7 8 9 T J Q K
Spades: A 2 3 4 5 6 7 8 9 T J Q K
Player3 plays QC.
Cards on the table:
Clubs: A 2 3 4 5 6 7 8 9 T J Q
Diamonds: 3 4 5 6 7 8 9 T J Q K
Hearts: A 2 3 4 5 6 7 8 9 T J Q K
Spades: A 2 3 4 5 6 7 8 9 T J Q K
Player4 plays 2D.
Player<1’s discards:
Player<1’s score: 48 + 0 = 48
Player<2’s discards:
Player<2’s score: 75 + 0 = 75
Player<3’s discards:
Player<3’s score: 58 + 0 = 58
Player<4’s discards: AD KC
Player<4’s score: 67 + 14 = 81
Player1 wins!
```

As you can see, this is identical to the result of the previous test showing that rage quit successfully turns all players into cpus

Testing CPU algorithm (test4.in)

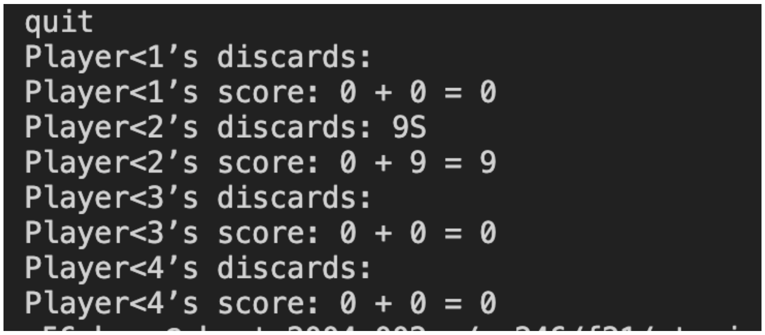
User reruns ./straights 200 with h, c, c, c for the player types and then player1 types deck and then plays 7S

As you can see, cpu2 has no valid plays so it discards 9S(their first card), cpu3 plays 7D (their first valid card), and cpu4 plays 7H (their first valid card)



```
e56zhang@ubuntu2004-002:~/cs246/f21/straights$ ./straights 200
Is Player1 a human (h) or a computer (c)?
h
Is Player2 a human (h) or a computer (c)?
c
Is Player3 a human (h) or a computer (c)?
c
Is Player4 a human (h) or a computer (c)?
c
A new round begins. It’s Player1’s turn to play.
Cards on the table:
Clubs:
Diamonds:
Hearts:
Spades:
Your hand: 8S AS 4C 6S 9D KS 4S 7C QS 3H 5D TD 7S
Legal plays: 7S
deck
8S AS 4C 6S 9D KS 4S 7C QS 3H 5D TD 7S
9S AD 4D 3D 6H 9C QD 2D JD 4H 5S AC JC
AH JS TH 5C 2S TC JH 2C 3S KD 7D 5H 8H
KC 3C 7H 9H QC QH TS 6C 8D 2H KH 8C 6D
play 7S
Player1 plays 7S.
Cards on the table:
Clubs:
Diamonds:
Hearts:
Spades: 7
Player2 discards 9S.
Cards on the table:
Clubs:
Diamonds:
Hearts:
Spades: 7
Player3 plays 7D.
Cards on the table:
Clubs:
Diamonds: 7
Hearts:
Spades: 7
Player4 plays 7H.
Cards on the table:
Clubs:
Diamonds: 7
Hearts: 7
Spades: 7
Your hand: 8S AS 4C 6S 9D KS 4S 7C QS 3H 5D TD
```

User plays 6S, and finds the algorithm persists. User quits and sees that the program exits without any issues and it displays the current score.



```
quit
Player<1’s discards:
Player<1’s score: 0 + 0 = 0
Player<2’s discards: 9S
Player<2’s score: 0 + 9 = 9
Player<3’s discards:
Player<3’s score: 0 + 0 = 0
Player<4’s discards:
Player<4’s score: 0 + 0 = 0
```