CS246 Final Project: Demo (Hydra)

Ethan Santos

2021-08-13

Welcome to the demo for my project. I will begin by briefly explaining the different options available for my program. Then, I will present various inputs to the program that will show its functionality.

1 Options: Enabling Modes

My program has three possible modes, selected with command-line arguments once the executable file "hydra" has been built:

1.1 Default

This mode is activated if **NO COMMAND LINE ARGS** are given:

```
$ ./hydra
```

This mode is random, and has the default functionality of Hydra.

1.2 Testing

This mode is activated by the **-testing** command-line argument:

```
$ ./hydra -testing
```

This mode is the same as the testing mode described in the given specifications.

1.3 Cheats

This mode is activated by the **-cheats** command-line argument:

```
$ ./hydra -cheats
```

This is an extension of the given testing mode that has several extra features. At the start of the program, after choosing the number of players, the following occurs:

i) The user chooses the number of cards in each player's starting draw pile

ii) The user chooses the number of active heads to start the game with

iii) For each active head:

   a) The user chooses the number of cards in the head

   b) The user chooses the topmost card of the head

Note that since this is an extension of testing mode, all features of testing mode are in this mode as well.

2 Demo

Now, we can proceed to demostrate the functionality of this program over several tests. The contents of the .in files are included in the pdf for convenience (note that a skipped line is equivalent to pressing enter).

2.1 Early Game

This test will confirm basic functionality:

```
$ ./hydra -testing < demo1.in
```

demo1.in:

```
2
3 H

5 H
1
K D
2 D
```

2.2 Late Game

This test shows endgame functionality, and some edge cases that occur when piles are empty:

```
$ ./hydra -cheats < demo2.in
```

note: make sure the cheats argument is being used! This mode is being used to "fast-forward" to an endgame scenario.

demo2.in:

```
3
10 2 1
5
2 Joker
1 3 D
10 K H
3 A D
2 4 H

2 C
1

2 D
0

3 D
2

10 H
4
```

2.3 Reserve

This test showcases the reserve:

```
$ ./hydra -testing < demo3.in
```

demo3.in:

```
4
5 C

6 D
0
1
6 D
8 S

2 S
0
3 D
0
5 D
0
```

2.4 Empty Draw

This test considers the case where the draw pile is empty, but the game is not won. An edge case test, as well as showcasing the discard pile being moved to the draw pile:

```
$ ./hydra -cheats < demo4.in
```

demo4.in:

```
2
4 1
1
10 5 S

6 H
1

A D
6 H
```

2.5 Joker

This test showcases Joker behaviour:

```
$ ./hydra -testing < demo5.in
```

demo5.in:

```
10
Joker

Joker
1
3
1
A
```

2.6 Invalid Inputs

This test shows invalid input behaviour:

```
$ ./hydra -testing < demo6.in
```

demo6.in:

```
2
a
0
1
2
1
```

```
c
C

Q H
d
-1
0
2
1
7 C
8 D

9 C
1
3
2
```

2.7 Win by Cutting

This test shows the case where a player wins by cutting a head:

```
$ ./hydra -cheats < demo7.in
```

demo7.in:

```
2
1
1
1
1
9 S

K H
1
9 S
9 S
```

2.8 Cheat Mode

This test further showcases the extra cheats mode:

```
$ ./hydra -cheats < demo8.in
```

demo8.in:

```
3
1 100 20
2
```

```
50 Joker
1 A C

A H
2
```