COMS 4115: DESCARTES

THURSDAY, DECEMBER 22ND

Eric Chao | Susan Fung | Jim Huang | Xiaocheng Shi
Overview: Motivation

- Design a simple card game language relying on a 52-card deck.
- Focus on younger/beginner programmers.

High Low Program
- Java Implementation – 185 lines
- Descartes Implementation – 80 lines
Language is structured to allow developer to focus on algorithms behind game.

Structure of language includes several types:
- Int – 0,1,2,3,4,5,6,7,8,9
- Bool – true | false
- String – “string”
- Card – S1, H2, DA, CQ, etc.

Static Typed Language
- Error handling is done during compilation process.
Overview: Some Built-in Functions

- printDeck()
  - Prints out the entire default deck.
- shuffleDeck()
  - Shuffles the entire deck.
- draw()
  - Draws one card from the top of the deck.
Basics 1: Ints, Bools

```c
int main()
{
    int i;
    i = 5;
    while (i > 0) {
        print(i);
        i--;
    }
    return i;
}
```

Prints: 543210

Basic 2: Strings, Functions

```c
string main() {
    string s1;
    s1 = "I am a string";
    return printString(printString(s1));
}
```

```c
string printString(string x;){
    string s;
    s = readStr();
    println(s);
    return s;
}
```

I am a string
I am a string
I am a string
Basics 3:

```c
string main() {
    string c;
    printDeck();
    shuffleDeck();
    printDeck();
    c = draw();
    printDeck();
    return c;}
```

This is a default deck:
```
C9HQC5S10H4D3H6SJDJCJHJH9SKH7D2S8D5S7D7SAH8C10D4D6C7H10H
S5H5C6CQHKCAD8HAS9C2C4DACKDK
```

This is the same deck shuffled:
```
H2S3HAD5CKQS5KGCJC5HJC10H10C9D3DKD8S10C2H5S2S6DQC6S9S4S7H
6H7DAH8CAD10SJD7C8C4D2HKC3D4
```

This is the rest of the deck:
```
S3HAD5CKQS5KGCJC5HJC10H10C9D3DKD8S10C2H5S2S6DQC6S9S4S7H4H
7DAH8CAD10SJD7C8C4D2HKC3D4
```

This was the first card drawn from the deck:
```
H2
```

(removed printing of comments, spacing)
Architecture

- Descartes Flow Diagram

- *.des input
- scanner.mll
- parser.mly
- ast.ml
- compile.ml
- *.pl output
- Perl Compiler
Roles & Responsibilities

- **Eric Chao**
  - scanner.mll, parser.mly, ast.ml, test automation

- **Susan Fung**
  - compile.ml, documentation

- **Jim Huang**
  - stdlibs, blackjack/highlow.des

- **Xiaocheng Shi**
  - compile.ml, stdlibs, highlow.des

- Test cases & documentation were a group effort.
Lessons Learned

- Follow up in person with professor/TA to receive feedback for LRM to ensure our design is correct.
- Have a good plan but also be flexible and willing to change it. Have a good fit for each group member within the group.
- It is better to split the project up with as few dependencies as possible.
- Start early!
Demo

- Descartes “Hello World”
  - Prints out the default.
  - Demonstrates deck functions – shuffle/draw
- High Low
  - User guesses next card.
- Blackjack
  - User plays against computer dealer