

# Session 2, January 18, 2011

## Some announcements:

### Problems?

- Problems with CS system accounts – [cshelp@uidaho.edu](mailto:cshelp@uidaho.edu)
- Problems getting/installing Unicon contact Dr. J.

### Reading Assignment

- Read Rabin book Chapter 1.2. We will be covering approximately one chapter of the book per lecture
- Read Chapters 1-6 of Dr. J's Unicon Games book, available through the main class webpage.
- Read [Why Minecraft Matters](#) – independent game development is still alive

### Game Competitions -- for what they're worth

All CS students like games, right? So there are hordes of student programming competitions with games as their theme. Here are two. The reason I don't officially endorse any of these is given after the hyphen.

- [Queue ICPC Challenge](#) - C, C++, Java, or Python
- [Imagine Cup Game Design Competition](#) - MS, MS, MS, or MS

### Rabin Book, Chapter 1.2

See highlights from [Games and Society](#) – while this is important matter, it's maybe not at the core of this course. Also note that the slides here are not the official slide for Rabin, 2<sup>nd</sup> edition. There are no official slides, so these are the 1<sup>st</sup> edition slides...

- Audience and demographics – we need to be able to understand the audience, getting away from the stereotypes.
- In 2003, 50% of Americans were identified as playing games, (its [different now](#))
- Adult Only titles seemed to make up a statistically insignificant portion in the 2003 ESA report

### Homework Assignment

See [Homework #2](#)

- Question about working in pairs – perhaps, but would need to surpass requirements
- DO NOT FORGET to label your code with your NAME
- Some type of readme will be helpful, to be able to understand how the game is to be played, user controls, etc.

- For part 2, we will be using Lord of the Rings Online, but it only runs on Windows, so make the appropriate arrangements if you plan on participating in the in-game class session.

## Highlights of [UG book](#) Chapters 1-6, cont'd

Reminder: The coding here is EASY, but you are supposed to supply corrections and ask nitpicky questions until you are very familiar with the syntax of our "very high level rapid game development language".

Ascii Art:

```

hangedperson := [
  "  o  ",
  "\\ | /",
  " \\|/ ",
  " /|\\ ",
  "/ | \\ "
]

```

Stopping a program with an error message:

```

if misses = 5 then
  stop("you lose! The word was ", word)

```

Checking if the guessed letter appears, and updating the string that keeps track of the correct part.

```

if find(letter, word) then
  every i := find(letter, word) do
    blanks[i] := word[i]
else misses := misses + 1

```

Open a window. Draw Stuff:

```

# open a file with "g" mode for graphics
w := open("hangman", "g", "size=400,400")
DrawCircle(w, 300, 50, 25)
DrawLine(w, 300,75, 300,150)

```

## Chapter 3: Dice Games

Roll 5 dice, Yahtzee style:

```

dice := [ ?6, ?6, ?6, ?6, ?6 ]
write("The dice are: ",
      dice[1], dice[2], dice[3], dice[4], dice[5])

```

Keeping some dice, re-rolling others:

```

write("Which dice values do you keep?")
keep := read()

# Reroll the dice
every i := 1 to 5 do {
  if not find(dice[i], keep) then
    dice[i] := ?6
  else keep[find(dice[i], keep)] := ""
}

```

Sorting the dice:

```
dice := sort(dice)
```

Scoring: upper half

```
if dice[1]=dice[2]=dice[3]=dice[4]=dice[5] then ...  
else if dice[1]=dice[2]=dice[3]=dice[4] then ...  
else if dice[1]=dice[2]=dice[3] then ...
```

## Brief Summary of Rules of Yahtzee

See <http://grail.sourceforge.net/demo/yahtzee/rules.html>

More generally: I welcome input where the Game Book needs additional material, and when it is missing something, I welcome questions. But when you need some info like this, please use Google and dig it up.

## Session 3, January 20, 2011

### Announcements

- Note taking – please get signed up via email with Dr. J. for your week.

### Mailbag

How do I declare "private" variables in Unicon classes?

Sorry, Unicon and many other dynamic languages use "bedroom door privacy", not "10-foot-walls-between-neighbor's-houses" privacy.

If someone enters "1 3 5 4 2" how do I get those into separate variables?

How about

```
L := []  
"1 3 5 4 2" ? {  
    while put(L, tab(many(&digits))) do  
        tab(upto(&digits))  
}
```

### Mention of [Pirate Resources](#) Page

- Range from kid stuff type knowledge, to popular fiction and historical accounts
- <http://www.talklikeapirate.com/> for battle banter on assignment #2
- Or [http://www.writersdigest.com/upload/images/Pirate\\_Insults.pdf](http://www.writersdigest.com/upload/images/Pirate_Insults.pdf)

## Rabin of the Day

Let's see if there are any interesting comments in [Chapter 1.3](#) on Ludology.

## Text Adventures and ASCII Art Games

The theme of Chapters 1-6 of the Unicon games book was: we can write computer programs which reasonably faithfully implement popular non-computer parlor games. How pretty we make them is subject to how much graphics or sound we invest, but the underlying computation might be complete and identical to the actions of the original game. For example, using a computer to play checkers with another human is really a game of checkers, even if no physical board or pieces are involved.

Most computer games are *simulations* of some imaginative situation. The computer both (a) allows more detailed simulations than players can manage using pencil and paper, and (b) frees the player from the need for a referee or game master.

## Text Adventures

See the Chapter 7 description of text adventure gaming and the show-and-tell book examples of interactive fiction.

[The ORIGINAL text adventure Infocom games \(Java applets\) Hamlet adventure \(original cia.icn\)](#) improved [cia2.icn](#)

Cia2.icn:

- Translated into Unicon from old (late 1970s/early 1980s) BASIC text based game (perhaps something like [this](#))
- Added object oriented slant with Unicon version
- Lots of string processing
- Adaptive help, perhaps at times offering help based on what you have in your possession or have accomplished

Steve Jackson wrote an [article](#) on writing interactive fictions which may be of interest.

## Turn-based Role-playing (text-based)

A new chapter 8 for the Unicon games book has a starting point for your homework assignment. See [piraduel.icn](#) and [ship.dat](#).

- Similar in spirit to [Rogue](#) or [NetHack](#)
- Interestingly, does not run in the console window, but opens its own window/GUI as it is
- Global objects for the player and world
- ! operator usage – generate all members of a list
  - In main procedure, used in an if statement, attempting to find some member of list that produces a true result
- Use as a jumping off place for homework 2?
- Examples of functional Unicon code?