CSE 13S

- Linux : Ubuntu 20.04.1
- Vim
- Git

On the nature of numbers

- numbers exist independtly of their representation
  - There are many ways to write the same number
- our method for writing numbers derives from Hindu Arabic numeral system
  - Aryabhata introduced positional notation, and Brahmagupta introduced zero
  - we write our numbers in base 10 since we have 10 fingers

Kinds of numbers

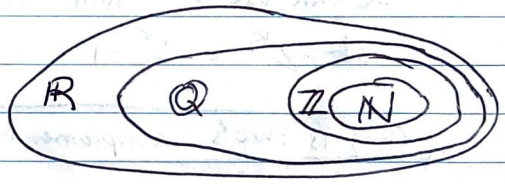
$\mathbb{N} = \{1, 2, 3, \ldots\}$ natural numbers

$\mathbb{Z} = \{\ldots -2, 1, 0, 1, 2 \ldots\}$ integers

$\mathbb{Q} = \{\frac{a}{b} : a, b \in \mathbb{Z}\}$ rational numbers

$\mathbb{R} = \mathbb{Q} \cup$ the irrational numbers

$\mathbb{C} = \mathbb{R} \cup$ the imaginary numbers



π, e, γ, φ are are irrational and are much more numerous than $\mathbb{Q}$

Computers can represent none of those sets

- computers do arithmetic in finite fields
  - represent finite amount of numbers
- A digit for a computer is usually called a bit
  - a bit is either 0 or 1

Specifying an integer in C

unsigned { short, long, long long } int

unsigned { char }

Name	size
char	usually 8 bits
short	usually 16 bits
int	at least 16 bits
long	at least 32 bits
long long	at least 64 bits

#include <stdint.h>

Binary arithmetic

addition
0+0 = 0
1+0 = 1
1+1 = 0 with carry out 1

Multiplication
0×0 = 0
1×0 = 0
1×1 = 0

Arithmetic in a finite field

- In a finite field we can define for every number its additive inverse.
  - if i is ~~supposed~~ a number in a finite field, and $\bar{i}$ its additive inverse
    $i + \bar{i} = 0$
  - suppose that we have k bits, then we can have every positive integer from 0 to $2^k - 1$
  - we can use one half and negative and other half is positive
    ~~2^k~~ -2^k to $2^k - 1$

This is two's complement arithmetic

Real Numbers

Real numbers (R) are
- continuous
- uncountably infinite

Include:
- Integers
- rational numbers
- irrational numbers

Floating point Numbers

float	single precision
double	double precision
long double	~~double~~ quad precision

big endian, little endian

- when we have an integer that requires multiple bytes, what is the address of the byte holding the LSB?
  - little endian means it is low address byte
  - big endian means it is the high address byte

Big Endian	Little Endian
12 | 34 | 56 | 78	78 | 56 | 34 | 12

Random Numbers

- True random numbers cannot be created using computers
- programs are inherently deterministic
- it has the advantage of repeatability, but
- it has the disadvantage of predictability

Arithmetic operators

* multiplication
/ division
% modulus (only integers)

Operator Precedence: (important)

SGT

If else: 2 conditions
switch: multiple cases

Loops

- while - most basic loop
- for loop: loop with fixed amount

do while - atleast once then checks condition

go-to statement -

Lab 1 design document notes

include:

- A general description of what your program will do and functions
- An in depth description of each function that you utilize within your program
- pseudo code
- pre-lab questions: these are mandatory

what makes a good design?

- An outline of what you want plan to do before you write your program.
- visualization of what your program intends to do
- A summarization of your final program and how you learned through it

assignment 1 notes

- $ rand
- while loop (for game) - checks to see if there is one player left
- array checks money
- helper functions
- void - returns nothing
  int - returns integer data type
- global variable = variable whole program
- local variable = variable for specific function