SFU SIMON FRASER UNIVERSITY
ENGAGING THE WORLD

CMPT 125 - Introduction to Computing Science and Programming II - Fall 2021

Lab 4. Binary Encoding
October 6

General notes

- Office hours:
  - Sepid: Tuesdays 10:00AM-11:00AM in ASB 9810
  - Kaumil: Wednesdays 11:30AM-12:30PM in ASB 9810

- Tutorials / Labs:
  - Wednesdays 9:30AM - 10:20AM, ASB 9838, Burnaby
  - Wednesdays 10:30AM - 11:20AM, ASB 9838, Burnaby
  - Wednesdays 3:30PM - 4:20PM, ASB 9838, Burnaby
  - Wednesdays 4:30PM - 5:20PM, ASB 9838, Burnaby

- Assignment grading:
  - Make sure to follow instructions carefully, even a single character can result compile error and a zero for your grade.

- Academic Honesty:
  - Do not use something that you can’t cite!

Outline

SFU

- Binary encodings of integers
  - Fixed width encoding
- Non integer arithmetic
  - Scientific Notation
  - Floating point encoding
- Bitwise operations

Binary encoding of integers

SFU

The values of digits in a number are positional:

Decimal numbers: $582 = 500 + 80 + 2 = 5*10^2 + 8*10^1 + 2*10^0$

Binary numbers: $10110 = 1*2^4 + 0*2^3 + 1*2^2 + 1*2^1 + 0*2^0$

Binary encoding of integers

SFU

The values of digits in a number are positional:

Decimal numbers: 582 = 500 + 80 + 2 = 5*102 + 8*101 + 2*100

Binary numbers: 10110 = 1*24 + 0*23 + 1*22 + 1*21 + 0*20

Exercises:

1. Convert 10011011 from binary to decimal.
2. Convert 29 from decimal to binary.
3. Write a program that get an integer in decimal and print its binary form.

Fixed width encoding

SFU

Simple data types are usually fixed in width:

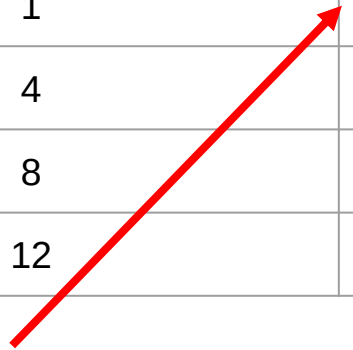
	bytes	
int	4	-2,147,483,648 to 2,147,483,647
long long int	8	-(2^63) to (2^63)-1
char	1	-128 to 127
float	4	
double	8	
long double	12	

Fixed width encoding

SFU

Simple data types are usually fixed in width:

	bytes	
int	4	-2,147,483,648 to 2,147,483,647
long long int	8	-(2^63) to (2^63)-1
char	1	-128 to 127
float	4	
double	8	
long double	12	



One of the digits is for the sign

Non integer arithmetic

Two common decimal numbers:

⅓ = 0.33333333333

⅔ = 0.6666666666

Cannot write the decimal with finite number of digits.

So we round them.

Scientific Notation

SFU

A convention to express numbers by their magnitude:

Speed of light = 2.99792458 x 108 m/s

One gigabyte = 1.073741824 x 109 bytes

⅓ = 3.33333333333 x 10-1

Same conventions are used for binary

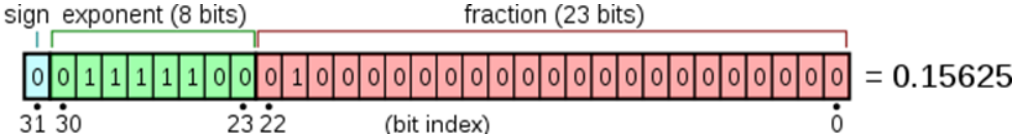
Floating point encoding

A float is composed of 32 bits:

1 bit for sign (0 – positive, 1 – negative)

23 bits for the significand (significant digits of the number)

1.b22b21…b0



b22 represents the digit of ½

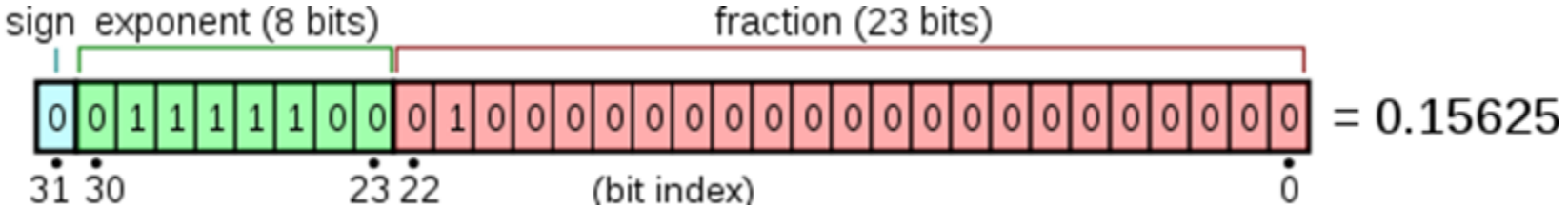
b21 represents the digit of ¼ …

8 bits for the exponent – ranges from -127 to 128

Range of the representation: 1.b22b21…b0 x 2(exp-127)

between ≈ 2128 ≈ 3.40 x 1038 and ≈ 2-127 ≈ 1.17 x 10-38

Floating point encoding



The number is (-1)sign x (1+ fraction) x 2exp-127

In this example:

1 x (1 + 1/4) x 2124-127

= 1.25 x 2-3

= 1.25/8 = 0.15625

Bitwise operations

SFU

Symbol	Operator
&	bitwise AND
|	bitwise inclusive OR
^	bitwise XOR (exclusive OR)
<<	left shift
>>	right shift
~	bitwise NOT (one's complement) (unary)

Bitwise operations

SFU

Exercises:

1. Assign the maximum and the minimum int to a variable using bitwise operations.

1. In the case of int type, what is the output of following equations?

a. (1<<31)-1 b. (-1<<3) c. (-1>>3) d. ~(-1)

3. Considering the tradeoff between memory and speed, what is the best way to keep a binary flag for an array with the minimum memory?

Bitwise operations

SFU

Exercises:

1. Assign the maximum and the minimum int to a variable using bitwise operations.

1. In the case of int type, what is the output of following equations?

a. (1<<31)-1 b. (-1<<3) c. (-1>>3) d. ~(-1)

3. Considering the tradeoff between memory and speed, what is the best way to keep a binary flag for an array with the minimum memory?

SFU

Programming exercises:

Please run the provided codes (bug, enc, large_int, unsigned_int_wraparound) and try to run and understand the outputs of each one.