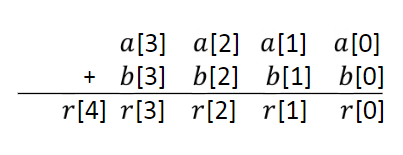
**COMP250 REVIEW**

**Lecture 1**

**Algorithm**: a sequence of instructions or operations for manipulating data to produce some result

Grade School Arithmetic:

1. Addition of 2 numbers:



- Use arrays to store each number

- For each column, add r[i] = a[i] + b[i] and carry value to next column

- The result array needs to have 1 more slot than max(a, b)

2. Multiplication of 2 numbers

- Slow: a\*b = add a times the number b

- Fast:

- Create 2D array and compute all rows in advance

- For each column, sum of the single digits in the row and add the carry

- Could instead use a running sum and add as we compute the rows

3. Division

- Slow: use num = q \* b + r. Subtract num by b, and add 1 to q, until r is smaller than b.

- Fast: long division

**Lecture 2**

Logarithms:

Modulo:

- Quotient remainder theorem: a = b\*q + r

- Remainder: a mod b = r

- Convention: find positive remainder

- However, in JAVA:

- In j=Java defined as:

- Addition:

- Multiplication:

Shift elements using mod:

- left shift n spots:

- right shift n spots:

**Lecture 3**

Base representation:

Base conversion: m, from decimal to base b

- divide m by b and prepend the remainder of division

- m = previous quotient

Binary Arithmetic:

- Carry when your sum is greater or equal to your base

- carry = # of times you can pull out base from sum

For all rows, and assign carry to r[n]

**Lecture 3**

Primitive type:

- predefined by the language

- named by a reserve keyword

In Java:

- Integer: byte, short, int, long

- Real num: float, double

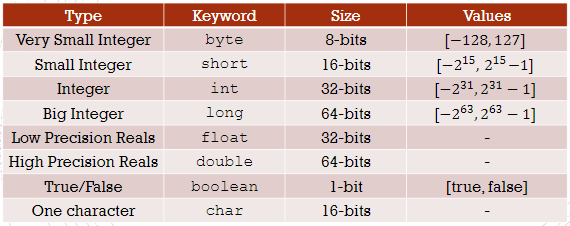
- Boolean

- Char

- One can represent values with bits

- One needs (round up) to represent different values

- One needs to represent a positive integer



Char

- single quotes

- single character

- A character set: an ordered list of character, **each character corresponds to a number** (i.e. Unicode)

Type Casting:

- Convert back and forth between variables of different types

- Explicit cast is not necessary when going from int -> double, but necessary in reverse!

- In general, widening (going to more bits) does not need explicit type conversion

- Char is more narrow than short, short can store negative values while char cannot, though they are of the same size