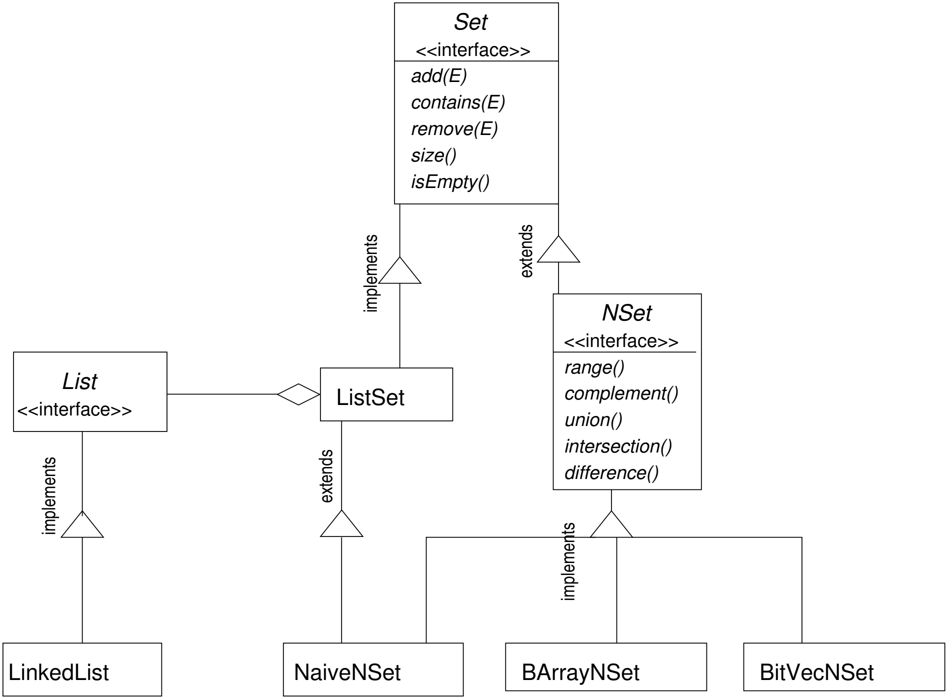
Chapter 3, Case Studies:

- Linear-time sorting algorithms (last week Monday and Wednesday)
- Disjoint sets and array forests (last week Friday)
- Priority queues (Monday and Today)
- N-sets and bit vectors (**Today**)
- (Start graphs Friday)

Today:

- Problem statement
- Abstractions and insights
- Project tips



Set

<<interface>>

add(E)

contains(E)

remove(E)

size()

isEmpty()

implements

extends

NSet

<<interface>>

range()

complement()

union()

intersection()

difference()

List

<<interface>>

ListSet

implements

extends

implements

LinkedList

NaiveNSet

BArrayNSet

BitVecNSet

$$\{1, 3, 4, 11\} \subseteq [0, 16)$$

F	T	F	T	T	F	F	F	F	F	F	T	F	F	F	F

Bitwise AND	&	result bit is set if both operand bits are set	0100110 & 1101011 0100010
Bitwise OR	|	result bit is set if at least one operand bit is set	0100110 | 1101011 1101111
Bitwise XOR	^	result bit is set if exactly one operand bit is set	0100110 ^ 1101011 1001101
Bitwise NEG	~	flip each bit of the operand	~ 1101011 0010100

Coming up: (all end-of-day)

Do **linear sorting** *project (suggested by this past Monday)*
Do **heaps and priority queue** *project (suggested by Mon, Feb 14)*
Do **bit vector and N-set** *project (suggested by Wed, Feb 23)*

Due **Today***:*
Take heap/pq quiz

Due **Thurs, Feb 10***:*
Read Section 3.4
Do Exercises 3.(27 & 28)
Take N-sets quiz

Due **Wed, Feb 23** *(but spread it out):*
Read Section 4.(1–3)
Do Exercises 4.(22-25).
Take graph quiz