Lecture 11: More Sets and Maps

07/18/22
Reminders

• A2 Resubmission due Wednesday 7/20 @ 11:59pm

• Optional review session – Today @ 1:10pm in GUG 220 (here)

• Seat assignments posted on Exams page

• Midterm in lecture on Friday!
Set ADT

- **set**: A collection of unique values (no duplicates allowed)
  - add, remove, contains
  - no indices

```python
set.contains("to")
true
```

set

"if"  "the"  "of"
"to"  "from"  "you"
"by"  "she"  "him"
"in"  "why"
Map ADT

- **map**: Holds a set of key-value pairs, where each key is unique
  - a.k.a. "dictionary"

- **basic map operations**:
  - **put**(key, value): Adds a mapping from a key to a value.
  - **get**(key): Retrieves the value mapped to the key.
  - **remove**(key): Removes the given key and its mapped value.
Which of these options will add the friendship to the map?

```java
// Option A
if (!friendNetwork.containsKey(name1)) {
    ...
} else {
    Set<String> friendGroup = friendNetwork.get(name1);
    friendGroup.add(name2);
    friendNetwork.put(name1, friendGroup);
}

// Option B
if (!friendNetwork.containsKey(name1)) {
    ...
} else {
    Set<String> friendGroup = friendNetwork.get(name1);
    friendNetwork.put(name1, friendGroup);
    friendGroup.add(name2);
}

// Option C
if (!friendNetwork.containsKey(name1)) {
    ...
} else {
    Set<String> friendGroup = friendNetwork.get(name1);
    friendGroup.add(name2);
}
```
Grammars
Languages and Grammars

• (formal) **language**: A set of words or symbols.

• **grammar**: A description of a language that describes which sequences of symbols are allowed in that language.
  - describes language **syntax** (rules) but not **semantics** (meaning)
  - can be used to generate strings from a language, or to determine whether a given string belongs to a given language
Backus-Naur Form (BNF)

• Backus-Naur Form (BNF): A syntax for describing language grammars in terms of transformation rules, of the form:

```
<symbol> ::= <expression> | <expression> ... | <expression>
```

• terminal: A fundamental symbol of the language.
• non-terminal: A high-level symbol describing language syntax, which can be transformed into other non-terminal or terminal symbol(s) based on the rules of the grammar.

• developed by two Turing-award-winning computer scientists in 1960 to describe their new ALGOL programming language