COP 3330 10/26/26

1) Exam Handback

2) Cur Grade = [(P1+P2+P3+P4+P5)/10 + (E1+E2)/5] /.55

A ≥ 80%, B = ~~65%~~ 60%, C = 45%

higher than usual # grades < 45%

→ 5 practice programs/week + COME TO CLASS

Takes time

3) Continue w/ Java API

- HashSet ✓
- HashMap
- PriorityQueue (Wed)

HashSet

add, remove + search for items in expected O(1) time. (Math - NO DUPLICATES IN A SET)

$\{1,3,5\} \cup \{2,3\} = \{1,2,3,5\}$

add
clear
contains
remove
size

HashSet<String> set = HashSet<String>();

HashMap

```
HashMap<String, Integer> map = new HashMap<>();
```

- String: key type
- Integer: value type

Set
- apple
- banana

Map
- apple → 2
- banana → 1
- kiwi → 3

containsKey - O(1)
containsValue → O(n) x (don't use!)
get - O(1)

keySet() - used to loop through keys

~~remove~~ put() → adds entry
remove()
size()

To give unique ids:

```
int idx = 0;
HashMap<String, Integer> nameMap =
        new HashMap<>();

for (int i = 0; i<n; i++) {
    String name = sc.next();
    if ( ! ~~contains~~
        nameMap.containsKey(name))
        nameMap.put(name, idx++);
}
```

Keys are unique
But values may not be.