Chapter 6, Hash tables:

- General introduction; separate chaining (last Friday)
- Open addressing (Today)
- Hash functions (Wednesday)
- Perfect hashing (Monday after next)
- Hash table performance (Wednesday after next)

Today:

- Review/finish hash table concepts
- Basic idea and example of open addressing
- Terminology, code, and invariant
- Probing strategies
- Deletion
Invariant (Class OpenAddressingHashMap)

1. *The table is not full; there exists* \( i \in [0, m) \) *such that* \( \text{table}[i] = \text{null} \).

2. *There are no breaks in the chain for any key in the table; for all* \( i \in [0, m) \) *such that* \( \text{table}[i] \) *contains key* \( k \),
   
   - *if* \( h(k) \leq i \), *then for all* \( j \in [h(k), i) \), \( \text{table}[j] \neq \text{null} \);
   - *if* \( i < h(k) \), *then for all* \( j \in [0, i) \cup [h(k), m) \), \( \text{table}[j] \neq \text{null} \).
Invariant (Loop of optimized remove in linear probing.)

For all positions $k \in (i, j)$, gap is the only position, if any, between its ideal place ($h(\text{keys}[k])$) and its actual place ($k$).
Coming up:

*Do Optimal BST project (suggested by this past Friday, April 8)*
*Do Open addressing with linear probing project (suggested by Monday, April 18)*

**Due Tues, Apr 12**
*Do practice problem, recreating separate chaining example*
*Read Section 7.3*
*Take quiz*

**Due Mon, Apr 18**
*Read Sections 7.(4 & 5)*
*(No practice problems or quiz)*