

Due Thu Nov 3 at the start of your lab section; Submit
 Server: class = cse2010, assignment = hw5SxIndividual
 Due Thu Nov 3 at the end of your lab section; Submit
 Server: class = cse2010, assignment = hw5SxGroupHelp
 x is 12, 34—your merged section number (or c for c
 submissions).

To improve the experience of customers, an online social network site would like to add a feature that can display and share a timeline of major events/photos of a user. The user might have many events, given a time range, how would you find/display events efficiently?

The goal of HW5 is to manage the timeline and allow the user to specify a time range to display/share his/her major events. To improve efficiency, your implementation uses a skip list that includes the following operations:

- `get(key)` // if `key` exists, return value associated with `key`; otherwise, return null
- `put(key, value)` // if `key` doesn't exist, add entry and return null; otherwise, replace value and return the old value
- `remove(key)` // if `key` exists, remove entry and return its value; otherwise, return null
- `floorEntry(key)` // return the entry with the largest `key` less than or equal to `key`; return null if no such entry exists
- `ceilingEntry(key)` // return the entry with the smallest `key` greater than or equal to `key`; return null if no such entry exists
- `subMap(key1, key2)` // return all entries with `key` such that $key1 \leq key \leq key2$

Use `getRandHeight()` in `FakeRandHeight.java` for `put(key, value)` Program files are on the course website. We will evaluate your submissions on `code01.fit.edu` so we strongly recommend you to test your programs on `code01.fit.edu`. To preserve invisible characters, we strongly recommend you to download and save, NOT copy and paste, input data files.

Input: Input is from the command-line arguments for `HW5.java`:

- filename of actions, each line has one of the following actions:
 - `DisplayEvent date`
 - `AddEvent date event`
 - `DeleteEvent date`
 - `DisplayEventsBetweenDates startDate endDate`
 - `DisplayEventsFromStartDate startDate`
 - `DisplayEventsToEndDate endDate`
 - `DisplayAllEvents`
 - `PrintSkipList`

For simplicity, dates are in MMDD format (MM is 01-12, DD is 01-31). You may assume the dates are unique (a more detailed timestamp would be unique in the real-world) and each event is at most 100 characters. You may assume the skip list can have a height of at most 10. Sample input is on the course website.

Output: Output goes to the standard output (screen), each line has a result for the corresponding action:

- `DisplayEvent date event/none`
- `AddEvent date event success`
- `DeleteEvent date success/noDateError`
- `DisplayEventsBetweenDates startDate endDate date1:event1 ... or none`
- `DisplayEventsFromStartDate startDate date1:event1 ... or none`
- `DisplayEventsToEndDate endDate date1:event1 ... or none`
- `DisplayAllEvents date1:event1 ... or none`
- `PrintSkipList`
`(Sh) empty`
`...`
`(S1) date1:event1 ...`
`(S0) date1:event1 ...`

Sample output is on the course website.

Submission: Submit `HW5.java` that has the main method, and other program files. Submissions for Individual and GroupHelp have the same guidelines as HW1.

Note the late penalty on the syllabus if you submit after the due date and time as specified at the top of the assignment.