One-Day Assignment 6 – algorithmic solution

Kattis’s Quest

One Day Assignment 6 – Algorithm

- When a new quest is added, add it to your data structure – (1)
- When a query occurs:
  - While true:
    - quest = get_suitable_quest(remaining_energy) – (2)
    - if quest is null, break
    - answer += quest.quest_gold_reward
    - remaining_energy -= quest.quest_energy_cost
    - Remove quest from data structure – (3)
  - Output answer (should use a **long** instead of int)

One Day Assignment 6 – Representation

- The issue of TreeSet not supporting duplicate elements is the main concern for this problem
- The following slides document several possible ways to handle the 3 different parts of the algorithm in the previous slide (indicated with a (1), (2), or (3))
- In the following slides, *energy* refers to the energy cost of the quest, *reward* refers to the gold reward, and *remaining_energy* refers to the amount of energy left

One Day Assignment 6 – Representation

- Option 1: Use TreeSet<IntegerTriple>
- IntegerTriple contains 3 values:
  - First value: energy cost of the quest
  - Second value: gold reward of the quest
  - Third value: special unique id (to ensure we can support duplicates)
- (1) – Add new IntegerTriple (*energy*, *reward*, id++)
- (2) – floor() of IntegerTriple (*remaining_energy*, INF, INF), where INF is a large integer
- (3) – Remove the IntegerTriple found in step (2)

One Day Assignment 6 – Representation

- Option 2: Use TreeMap<Integer, PriorityQueue<Integer>>
- The key Integer contains 1 value:
  - Energy cost of the quest
- The Integer value in PriorityQueue (max heap) contains 1 value:
  - Gold reward of the quest
- (continued in next slide)

One Day Assignment 6 – Representation

- (cont. from previous slide)
- (1) – If *energy* is not present in the TreeMap, add a new entry mapping *energy* -> new PQ.
  Regardless of the above, find *energy* in the TreeMap, and add *reward* to the associated PQ.
- (2) – floorEntry() of *remaining_energy*, and peek from the associated PQ
- (3) – Pop from the PQ in (2). Remove entry from the TreeMap if the PQ is empty