CS2030 Lab 2

Cruise Loaders

Qi Ji (qiji@u.nus.edu)

12th February 2019

Task

- Given a cruise schedule on a single day, print stats.
- 2 cruise types
  - normal
  - big
- 2 loader types
  - normal
  - recycled (less useful)
- input: number of cruises, list of cruises with arrival times

Levels

1. read cruises in and print them out
2. print the loader schedule
   - Proceed with the basic program first: no big cruises/recycled loaders yet.
3. introduce Big Cruises
   - Inheritance and polymorphism.
4. introduce Recycled Loaders
   - See above
5. pretty-printing

Level 1: read stuff in

Input format:

- Line 1: `numberOfCruises` :: `int`,
  $0 \leq$ `numberOfCruises` ≤ 30
- Line 2 to $n + 1$: `cruiseCode arrivalTime`
  `cruiseCode` :: `String`
  `arrivalTime` :: `int`

Output format:

- `cruiseCode@arrivalTime` string representation of a cruise
  - `arrivalTime` must be formatted as a 24h time

Level 2: loader schedule

- Loader spends 30 minutes loading a normal cruise.
- Maintain an inventory of loaders.
- Get next free loader or purchase new loader if no freeloaders.

Loader IDs

Consider using a `static int` as counter.

Level 3: Big cruises

- Cruise is big if its code starts with 'B'.
- Needs 2 loaders.
- Service time: 60 minutes.
- Recommended approach: inheritance.

  works but not recommended: `public boolean isBig`

Level 4: Recycled loaders

- Takes a 60 min break **after** each service.
- Every third loader is recycled.
  - (normal) Loader 1
  - Loader 2
  - Recycled loader 1
  - Loader 3
  - ...

Level 5: Pretty-printing

- Each line is 36 characters long.
- Match the given output.

Note

- Number of cruises does not include big cruises.
- Number of loaders does not include recycled loaders.