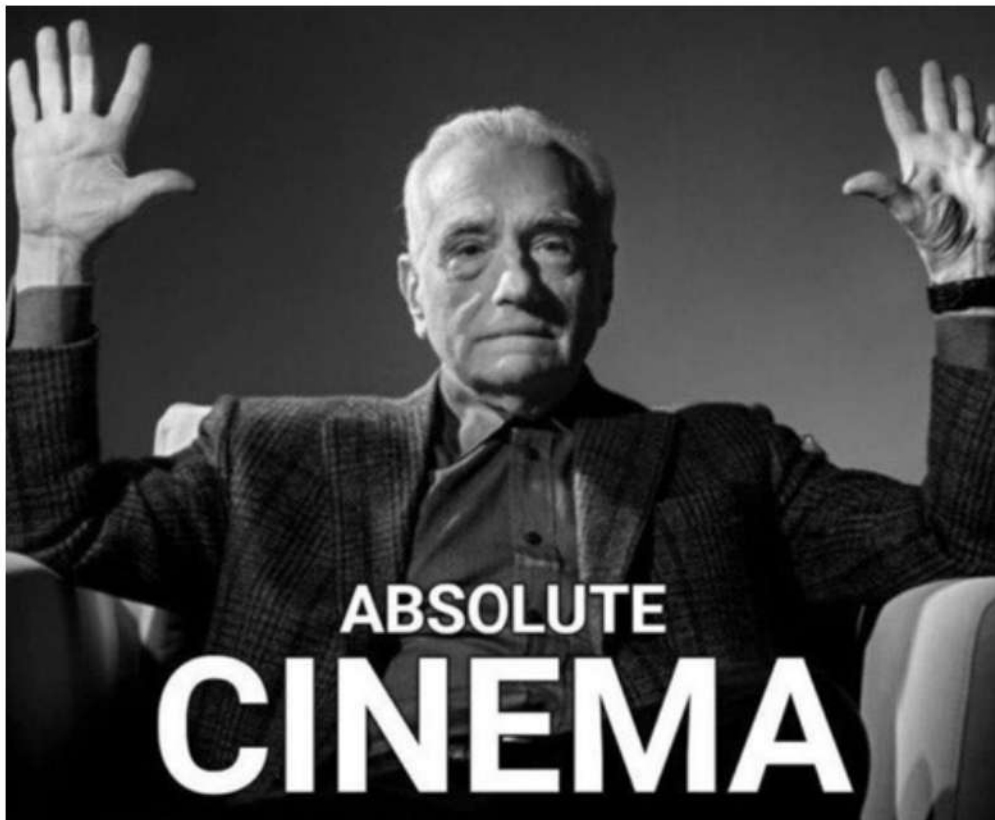


Sofita Si Kritikus Film!



Credit: [Martin Scorsese](#)

Deskripsi

Di tengah kesibukan kuliah, Sofita menyempatkan diri untuk mengikuti UKF SINTAKS. Ia sering kali menghabiskan akhir pekannya dengan menonton film bersama teman-teman SINTAKS. Dia iseng mengutarakan pendapatnya mengenai film yang ditonton pada situs SDA Tomatoes, sebuah situs web yang menyediakan ulasan dan penilaian film. Ternyata, ulasannya banyak dibaca oleh pengguna lain dan ia menjadi kritikus ternama yang lebih terkenal dibandingkan SZA.

Seiring berjalannya waktu, popularitas Sofita semakin meningkat. Kini, ia mendapatkan kepercayaan sebagai admin di SDA Tomatoes, yang memungkinkannya menambahkan film baru untuk diulas. Film yang ada di sistem memiliki 2 informasi utama, yaitu **ID Film**

(dimulai dari 0) dan **Jumlah Vote Film**, yang keduanya berupa bilangan bulat. Film yang ada di dalam sistem diurutkan dari **jumlah vote yang paling besar ke yang paling kecil**. Apabila ada Film dengan jumlah vote yang sama, maka film akan diurutkan dari **ID Film yang paling kecil ke yang paling besar**.

Dengan wewenang barunya, Sofita bisa memberikan kritik pedas terhadap film-film paling ternama saat ini. Para pengunjung situs juga mulai meminta Sofita untuk memberikan rekomendasi film terbaik menurutnya saat ini. Tak hanya itu, Sofita juga memiliki kemampuan untuk membandingkan dua film sekaligus, memberikan analisis mendalam mengenai kelebihan dan kekurangan masing-masing.

Selain itu, ada beberapa *query* yang dapat dilakukan oleh Sofita selama menjadi admin di SDA Tomatoes:

- **T [VOTE]**
 - Memasukkan film baru ke dalam sistem dengan vote awal sejumlah [VOTE].
- **K [DOWN_VOTE]**
 - Mengurangi jumlah vote film terbaik sebesar [DOWN_VOTE].
- **R [NUMBER_OF_FILM]**
 - Menampilkan beberapa ID film terbaik sebanyak [NUMBER_OF_FILM] film.
- **B [ID1] [ID2]**
 - Membandingkan film dengan [ID1] dan [ID2], lalu menampilkan ID yang memenangkan perbandingan yaitu **ID memiliki vote lebih banyak**.
 - Jika jumlah vote film sama, maka bandingkan ID-nya, film dengan ID **lebih kecil akan memenangkan perbandingan**.
 - Jumlah vote film yang **memenangkan perbandingan akan dibagi 2** dari jumlah vote sebelumnya. (Jika hasilnya bilangan desimal, maka dibulatkan ke bawah.)

Format Masukan

- Baris pertama berisi sebuah bilangan bulat N yang menyatakan jumlah film yang ada di SDA TOMATOES.
- N baris selanjutnya berisi vote V_i .
- Baris selanjutnya berisi sebuah bilangan bulat Q yang menyatakan banyaknya *query*.
- Q baris selanjutnya berisi *query* yang dilakukan oleh Sofita.

Format Keluaran

- Query T
 - Print **ID Film** yang memiliki voting paling banyak dan **jumlah vote film** tersebut **dipisahkan oleh sebuah spasi**.
- Query K
 - Print **ID Film** yang memiliki voting paling banyak dan **jumlah vote film** tersebut **dipisahkan oleh sebuah spasi**.
- Query R
 - Print **ID Film** sebanyak [NUMBER_OF_FILM] yang dipisahkan oleh sebuah spasi dimulai dari **jumlah voting paling banyak ke paling sedikit**.
 - Apabila [NUMBER_OF_FILM] lebih banyak dari jumlah film, print **-1**.
- Query B
 - Print **ID Film** yang memiliki vote lebih banyak dibandingkan lawannya.
 - Jika jumlah vote sama, print **ID Film** yang memiliki **ID terkecil**

Batasan

- $0 \leq N \leq 10^5$
- $1 \leq V_i \leq 2 \times 10^5$
- $1 \leq Q \leq 10^5$
- Pada kueri R: $1 \leq \text{NUMBER_OF_FILM} \leq 10$

Informasi Tambahan

Dilarang menggunakan priority queue bawaan java. Implementasi heap harus dilakukan secara mandiri.

Contoh Masukan 1

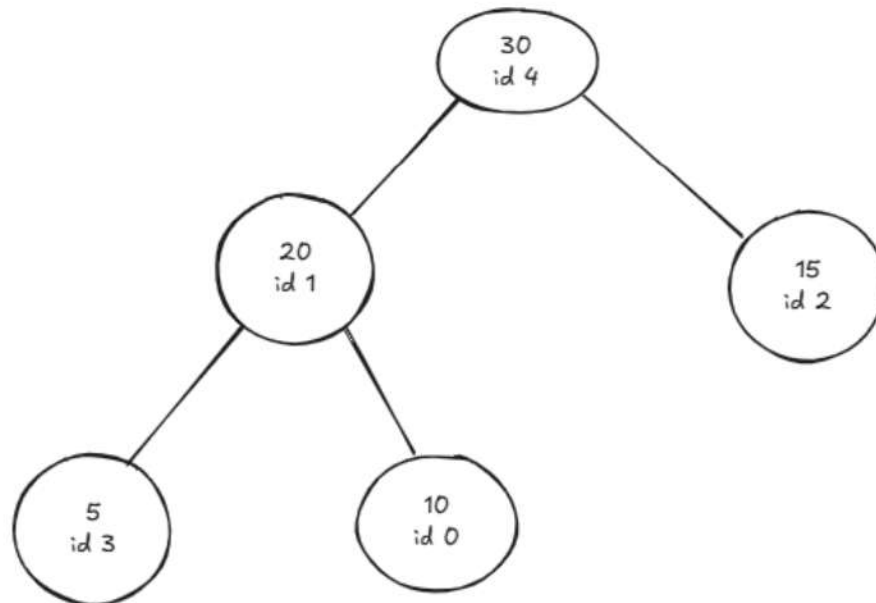
```
5
10
20
15
5
30
9
T 25
K 10
R 3
B 0 1
K 5
B 4 2
T 1
R 5
T 1
```

Contoh Keluaran 1

```
4 30
5 25
5 1 4
1
4 20
4
5 20
5 2 0 1 4
5 20
```

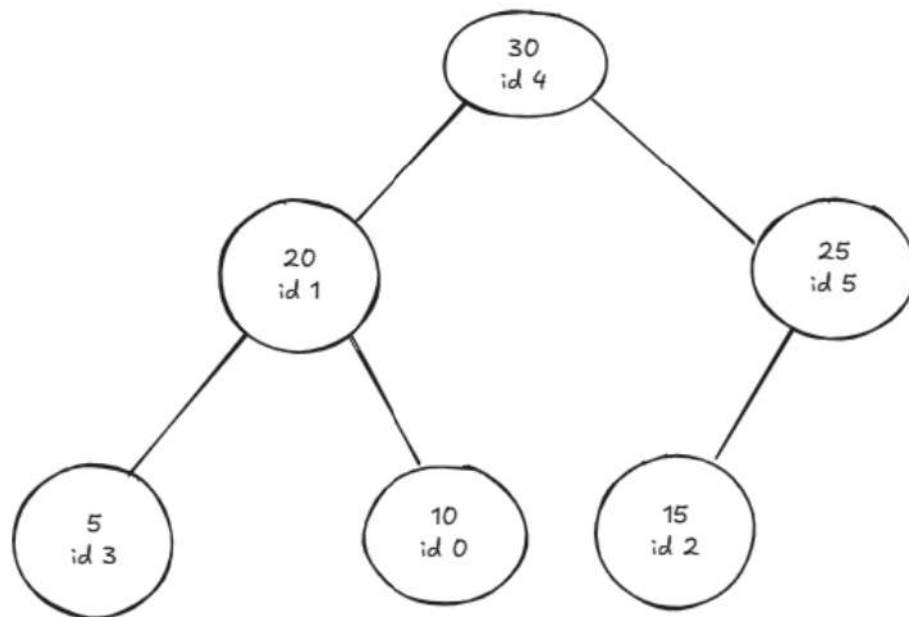
Penjelasan Contoh 1

1. Kondisi awal



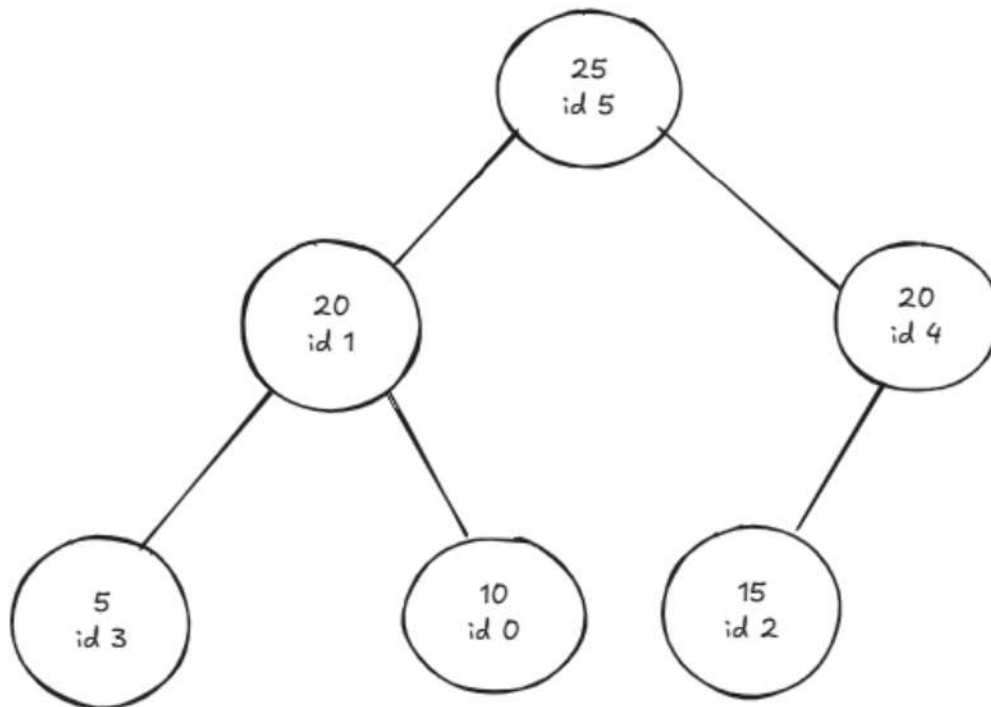
2. T 25

- Film baru dengan score 25 dan id 5 ditambahkan
- Berikut merupakan heap setelah ditambahkan



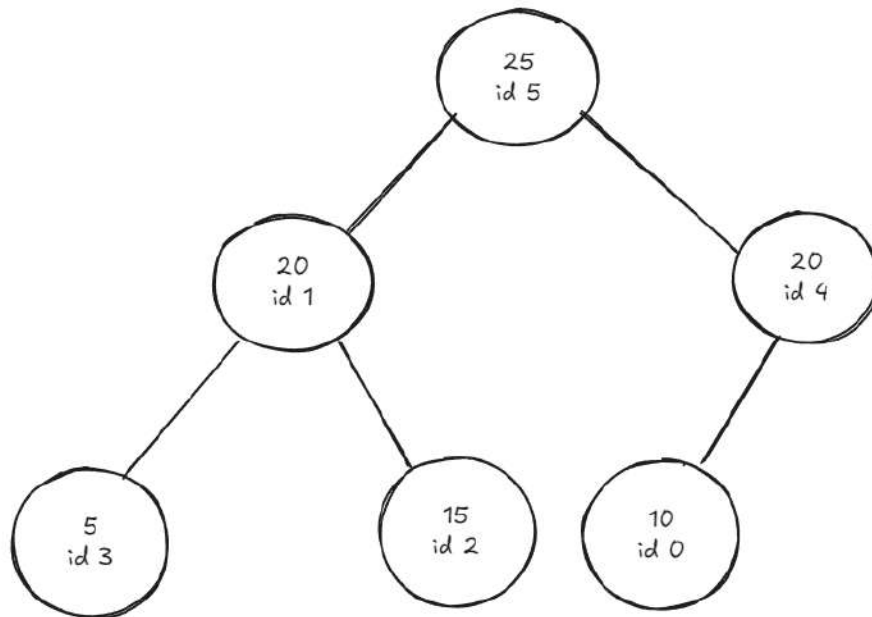
3. K 10

- Query kritik akan mengurangi score film teratas, pada saat ini skor teratas ada pada film dengan id 4 sehingga score-nya akan dikurangi 10 menjadi 20
- Berikut adalah kondisi heap setelah query ini



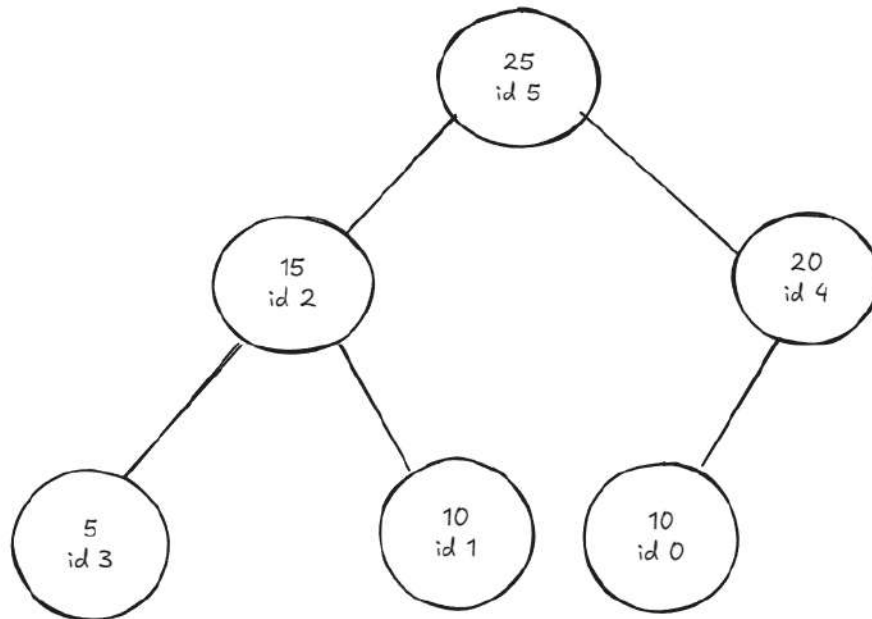
4. R 3

- Query rekomendasi dengan $n=3$ akan mengembalikan 3 film teratas pada heap
- Rekomendasi pertama adalah film teratas pertama yaitu film dengan id 5, sehingga output pertama adalah 5
- Film teratas kedua adalah film dengan id 1, sehingga output kedua adalah 1
- film teratas ketiga adalah film dengan id 4, sehingga output ketiga adalah 4
- Output query ini adalah semua idnya
- Berikut adalah kondisi akhir heap setelah query



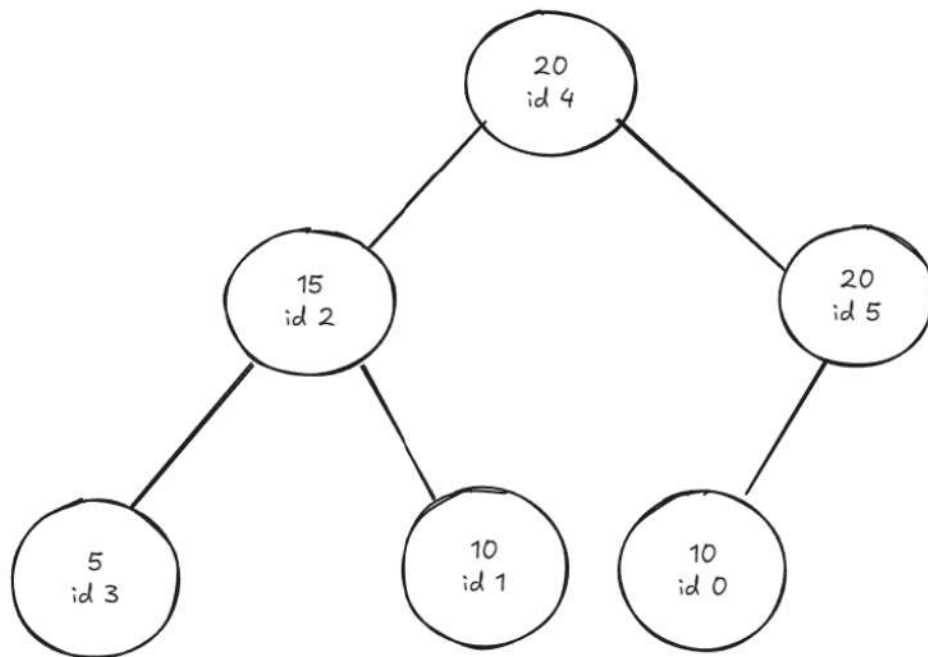
5. B 0 1

- Output akan mengeluarkan 1 karena film dengan id 1 memiliki score lebih besar daripada film dengan id 0. Karena film dengan id 1 menang, maka score miliknya akan dibagi 2 menjadi 10
- Dibawah ini merupakan kondisi akhir heap



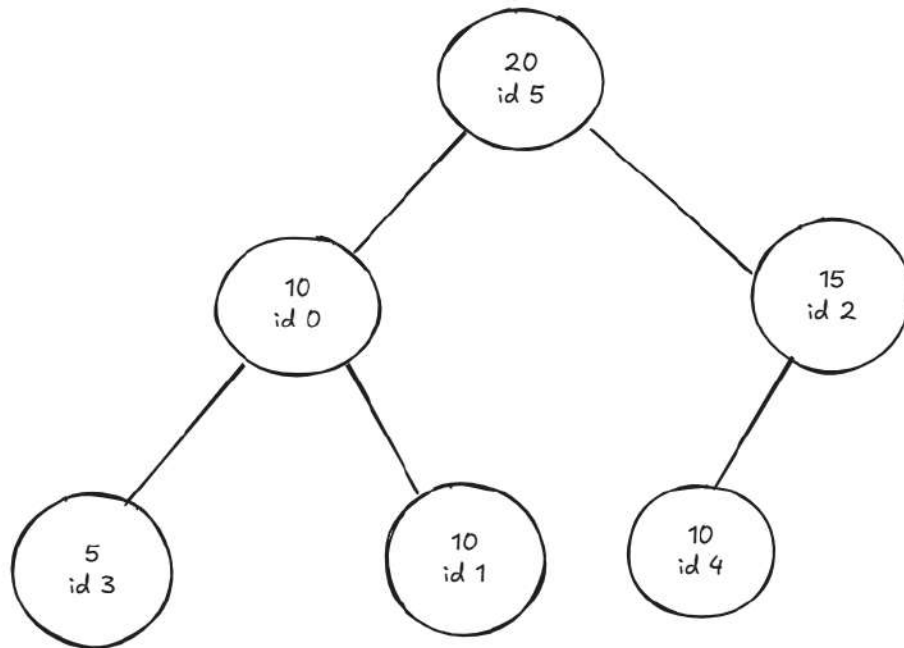
6. K 5

- Film dengan Vote terbesar saat ini adalah film dengan ID 5 dengan vote sebesar 25. Film dengan ID 5 akan dikurang dengan 5 menjadi 20
- Karena Film ID 5 dan ID 4 memiliki skor yang sama, maka prioritasnya adalah film dengan ID 4 karena memiliki ID yang lebih kecil sehingga outputnya adalah 4 20
- Berikut Kondisi akhir heap



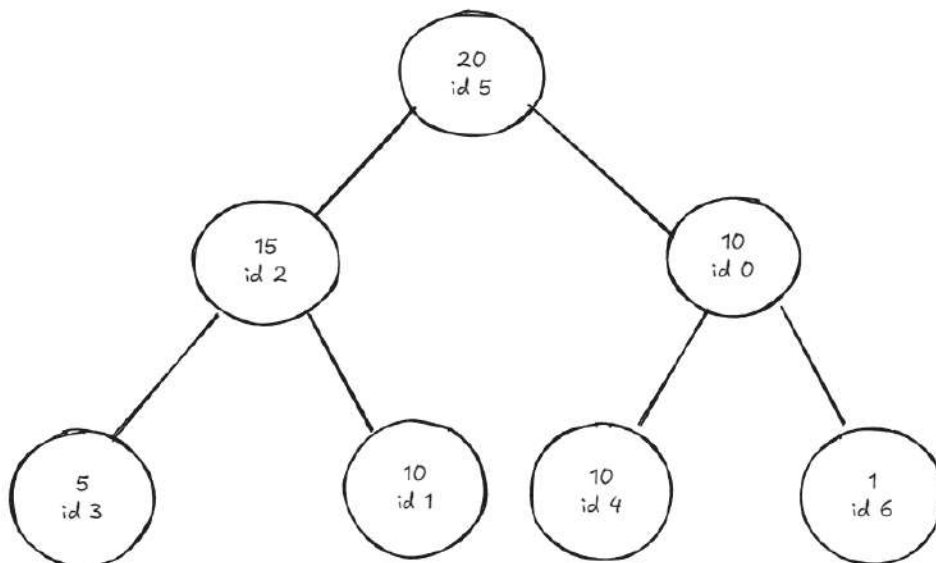
7. B 4 2

- Film dengan ID 4 memiliki vote sebesar 20 dan film dengan id 2 memiliki vote sebesar 15. Karena vote film dengan ID 4 lebih besar, maka outputnya adalah 4 dan jumlah vote milik film ID 4 akan dibagi 2
- Berikut kondisi akhir heap



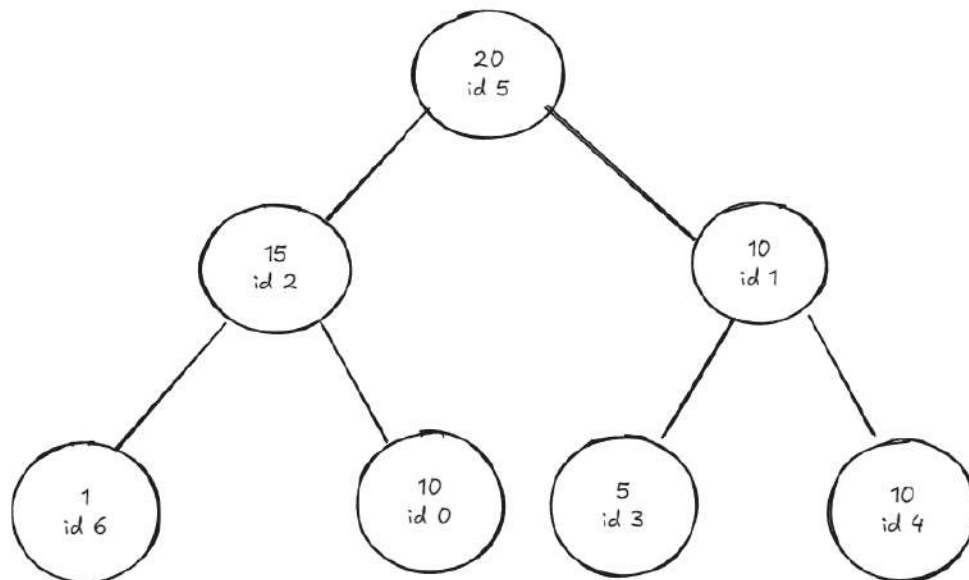
8. T 1

- Film baru dengan score 1 dan id 6 ditambahkan
- Berikut merupakan heap setelah ditambahkan



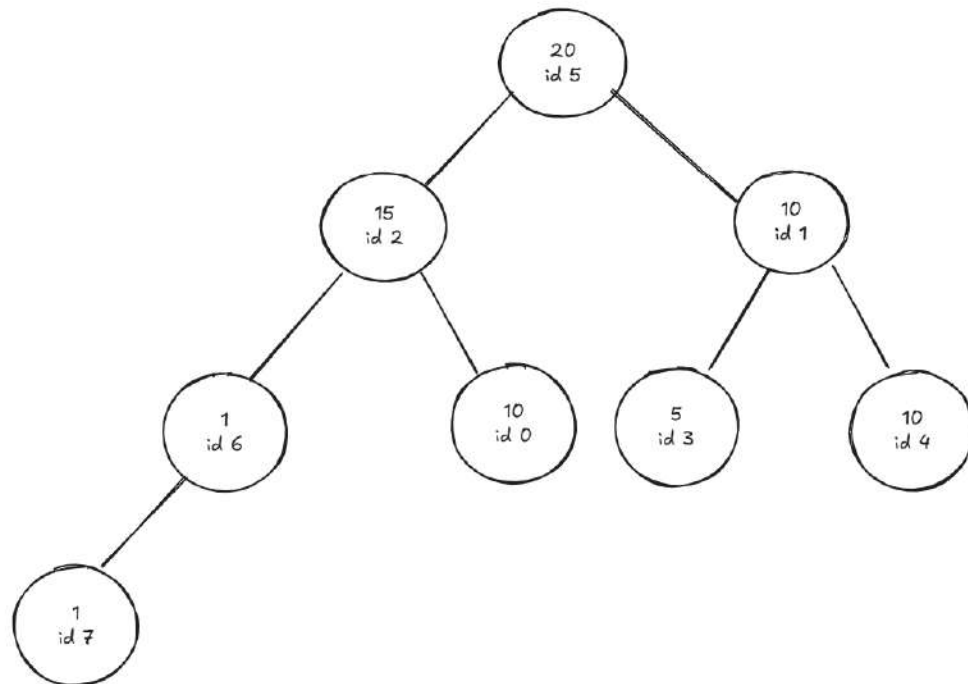
9. R 5

- Query rekomendasi dengan $n=5$ akan mengembalikan 5 film teratas pada heap
- Rekomendasi pertama adalah film teratas pertama yaitu film dengan id 5, sehingga output pertama adalah 5
- Film teratas kedua adalah film dengan id 2, sehingga output kedua adalah 1
- Film teratas ketiga adalah film dengan id 0, sehingga output ketiga adalah 0
- Film teratas keempat adalah film dengan id 1, sehingga output keempat adalah 1
- Film teratas kelima adalah film dengan id 4, sehingga output kelima adalah 4
- Output query ini adalah semua idnya
- Berikut adalah kondisi akhir heap setelah query



10. T 1

- Film baru dengan score 1 dan id 7 ditambahkan
- Berikut merupakan heap setelah ditambahkan



Informasi Tambahan Test-case

Pada 20% test-case berlaku query T, K

Pada 30% test-case berlaku query T, K, R

Pada 50% test-case berlaku query T, K, R, B