

Latihan Soal: ADT Queue

IF2110/IF2111 – Algoritma dan Struktur Data
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Soal 1 – Circular Buffer

- a. Definiskan struktur data yang merepresentasi Queue bertipe `ElType` yang terdiri atas `<id: integer, cost: integer>` dalam bentuk *circular buffer*, dengan alokasi statik maksimum 100 elemen, dan menyimpan informasi indeks head dan count (banyaknya elemen dalam Queue)
- b. Buatlah function `isFull`
- c. Buatlah procedure `enqueue`
- d. Buatlah procedure `dequeue`

Soal 1

function isFull (q: Queue) → boolean
{mengirim true jika q penuh}

procedure enqueue (input/output q: Queue, input val: ElType)
{Proses: menambahkan val pada q sebagai Tail baru}
{IS: q mungkin kosong, q tidak penuh}
{FS: val menjadi Tail baru dengan mekanisme *circular buffer*}

procedure dequeue (input/output q: Queue, output val: ElType)
{Proses: menyimpan nilai Head q ke val dan menghapus Head q}
{IS: q tidak kosong}
{FS: val adalah nilai elemen Head, Head “bergerak” dengan mekanisme *circular buffer*. q mungkin kosong}

Soal 2 – Round Robin

Pandanglah Queue pada soal nomor 1 sebagai antrian pekerjaan dengan id adalah nomor identifikasi pekerjaan dan “cost” adalah cost waktu penyelesaian pekerjaan (*time cost*).

Dengan memanfaatkan queue pada soal nomor 1, buatlah procedure **roundRobin** yang memproses Queue secara Round Robin, yaitu memproses dengan waktu terbatas T :

- Jika elemen pada HEAD memiliki $\text{cost} \leq T$, elemen tersebut dihapus dari Queue.
- Jika elemen pada HEAD memiliki $\text{cost} > T$, maka elemen tersebut dihapus dari Queue **dan** disisipkan kembali sebagai Tail dengan cost yang berkurang sebesar T .

Soal 2 – Round Robin

```
procedure roundRobin (input/output q: Queue, input t: integer)  
{Proses: memproses elemen antrian q secara round robin}  
{IS: q tidak kosong, t adalah waktu yang tersedia untuk memproses setiap elemen}  
{FS: elemen e pada posisi HEAD dihapus dari q.  
    Jika  $\text{cost } e \leq t$  maka ditampilkan “<id> telah selesai diproses”.  
    Jika  $\text{cost } e > t$  maka e disisipkan kembali sebagai tail q  
    dengan cost berkurang sebesar t }
```

Soal 3 – Priority Queue

- a. Dengan memodifikasi Queue alternatif 2 pada slide materi kuliah, definisikan (algoritmik) struktur data yang merepresentasi Queue yang menggambarkan antrian pekerjaan (job shop). Setiap elemen Queue bertipe EType yang terdiri atas $\langle id: \text{integer}, cost: \text{integer} \rangle$. id menunjukkan nomor identifikasi unik dari pekerjaan yang dikelola Queue, dan elemen Queue terurut membesar berdasarkan “cost” waktu memproses pekerjaan.
- b. Buatlah prosedur enqueue
- c. Buatlah prosedur dequeue

Soal 3 – Priority Queue

procedure enqueue (input/output q: Queue, input val: ElType)

{Proses: menambahkan val sebagai elemen baru di q, dengan memperhatikan lamanya waktu pekerjaan tsb dapat diselesaikan, yaitu pekerjaan yang lebih cost diletakkan lebih akhir. Jika ada 2 pekerjaan yang cost waktunya sama, pekerjaan terakhir yang baru datang disisipkan lebih belakang}

{IS: q mungkin kosong, q tidak penuh}

{FS: val menjadi elemen q yang baru dengan urutan waktu pekerjaan membesar}

procedure dequeue (input/output q: Queue, output val: ElType)

{Proses: menyimpan IDX_UNDEFa head q pada val dan menghapus head dari q}

{IS: q tidak kosong}

{FS: elemen pada HEAD dihapus, dan disimpan nilainya pada val}