



Program Studi Teknik Elektro ITB

Nama Kuliah (Kode) : Praktikum Pemecahan Masalah dengan C (EL2208)
Tahun / Semester : 2022-2023 / Genap
Modul : 7 - Stacks and Queues
Hari, Tanggal Praktikum : Rabu, 29 Maret 2023

Naskah Soal Praktikum

Pembuat Naskah: Kelvin Sutirta, Muhammad Daffa Rasyid

Ketentuan:

1. Kerjakanlah satu dari dua soal berikut pada *template repository* yang Anda peroleh ketika mengambil *assignment* di GitHub Classroom praktikum!
2. *Commit* yang dilakukan setelah sesi praktikum berakhir tidak akan dipertimbangkan dalam penilaian.
3. *Header* setiap *file* harus mengikuti format yang telah disediakan pada *file template repository*. *Header* yang tidak mengikuti format tersebut tidak akan dinilai.
4. Buku catatan laboratorium yang berisi *flowchart* dan *data flow diagram* dari solusi yang anda buat dikumpulkan ke praktikum.stei.itb.ac.id paling lambat pukul 11.00 WIB dua hari kerja setelah sesi praktikum. Tulis pula alasan anda memilih mengerjakan soal yang anda kerjakan dan penjelasan/rancangan dari algoritma yang Anda gunakan!
5. Solusi soal pertama dan kedua harus dapat dikompilasi dengan perintah `make main` dan menghasilkan *file executable* dengan nama `main`.
6. Bila diperlukan, sesuaikanlah isi *Makefile* yang tersedia pada *template repository* untuk memenuhi syarat kompilasi dan *file* keluaran di atas!

Soal 1

Pak Satria yang akrab disapa Bang Sat adalah seorang pengajar di bimbingan belajar *programming*. Beliau juga dikontrak oleh sebuah *startup* Jooxtify untuk membuat aplikasi *music player*. Akan tetapi, jadwal mengajar beliau sangat padat sehingga beliau memutuskan untuk menawarkan proyek tersebut kepada mahasiswa Teknik Elektro ITB. Anda bermimpi bahwa Anda lah yang dipilih oleh Bang Sat. Bantulah Bang Sat untuk menyelesaikan proyek aplikasi *music player*.

Salah satu fitur dari *music player* adalah *playlist* Anda diminta membuat suatu program C yang berfungsi untuk menampilkan *playlist* serta lagu yang sedang diputar dengan menggunakan konsep **Stack** atau **Queue**. Berikut spesifikasinya.

1. Pada saat program pertama kali dijalankan, ditampilkan *playlist* kosong dengan pesan (Playlist kosong! Tambahkan lagu!). Kemudian program meminta masukan perintah dari pengguna/
2. Masukan perintah berupa karakter, sebagai berikut.
 - ‘+’ untuk menambahkan lagu ke dalam *playlist* dengan format judul_lagu,penyanyi
 - ‘-’ untuk menghapus lagu yang paling terakhir ditambahkan dari *playlist*
 - ‘>’ untuk memutar lagu selanjutnya
 - ‘<’ untuk memutar lagu sebelumnya
 - ‘R’ untuk memulai kembali lagu dari urutan pertama *playlist*
 - ‘E’ untuk keluar dari program

Selain karakter diatas, input dinyatakan tidak valid dengan pesan “Masukan Perintah Tidak Valid!”

3. Untuk memutar lagu untuk pertama kalinya, digunakan perintah ‘>’ sehingga muncul tanda panah (->) yang menunjukkan lagu paling pertama dimasukkan ke dalam *playlist*.
4. Jika telah mencapai lagu paling terakhir, perintah ‘>’ akan membuat lagu yang diputar menjadi paling pertama. Sebaliknya, jika lagu paling pertama yang diputar, perintah ‘<’ akan membuat lagu yang diputar menjadi paling terakhir.

Hint : *Parsing* masukan nama lagu dan pemain dapat memanfaatkan **strtok**. Telah disediakan **struct song** untuk menyimpan judul lagu dan penyanyi, Anda diperbolehkan menambah variabel ke dalam *struct*. Dalam mencetak *playlist* ke layar, Anda mungkin memerlukan fungsi untuk membalikkan isi dari *stack*.

Contoh Eksekusi Program (garis bawah menandakan input)

```
#1
-*_-*_-*_-*_Jooxtify-*_-*_-*_-*_
Now Playing:
Playlist kosong! Tambahkan lagu!

Masukkan Perintah: j
Masukan Perintah Tidak Valid!
Now Playing:
Playlist kosong! Tambahkan lagu!

Masukkan Perintah: k
Masukan Perintah Tidak Valid!
Now Playing:
Playlist kosong! Tambahkan lagu!

Masukkan Perintah: +
Masukkan Nama Lagu dan Penyanyi: Sayang,Via Telepon

Now Playing:
    Sayang oleh Via Telepon

Masukkan Perintah: >

Now Playing:
-> Sayang oleh Via Telepon

Masukkan Perintah: +
Masukkan Nama Lagu dan Penyanyi: Sepertinya Bosan,KhanDar

Now Playing:
-> Sayang oleh Via Telepon
    Sepertinya Bosan oleh KhanDar

Masukkan Perintah: .
Masukan Perintah Tidak Valid!
Now Playing:
-> Sayang oleh Via Telepon
    Sepertinya Bosan oleh KhanDar

Masukkan Perintah: >

Now Playing:
    Sayang oleh Via Telepon
-> Sepertinya Bosan oleh KhanDar

Masukkan Perintah: >

Now Playing:
-> Sayang oleh Via Telepon
    Sepertinya Bosan oleh KhanDar

Masukkan Perintah: <
```

Now Playing:
Sayang oleh Via Telepon
-> Sepertinya Bosan oleh KhanDar

Masukkan Perintah: E

#2

--_--_--_--Jooxtify--_--_--_--

Now Playing:
Playlist kosong! Tambahkan lagu!

Masukkan Perintah: +
Masukkan Nama Lagu dan Penyanyi: Naik,Ke Puncak Gunung

Now Playing:
Naik oleh Ke Puncak Gunung

Masukkan Perintah: +
Masukkan Nama Lagu dan Penyanyi: Tinggi Tinggi,Sekali Duakali

Now Playing:
Naik oleh Ke Puncak Gunung
Tinggi Tinggi oleh Sekali Duakali

Masukkan Perintah: >

Now Playing:
-> Naik oleh Ke Puncak Gunung
Tinggi Tinggi oleh Sekali Duakali

Masukkan Perintah: >

Now Playing:
Naik oleh Ke Puncak Gunung
-> Tinggi Tinggi oleh Sekali Duakali

Masukkan Perintah: =

Now Playing:
Naik oleh Ke Puncak Gunung

Masukkan Perintah: >

Now Playing:
-> Naik oleh Ke Puncak Gunung

Masukkan Perintah: =

Now Playing:
Playlist kosong! Tambahkan lagu!

Masukkan Perintah: E

#3

--_--_--_--Jooxtify--_--_--_--

Now Playing:

Playlist kosong! Tambahkan lagu!

Masukkan Perintah: \pm

Masukkan Nama Lagu dan Penyanyi: Satu Tambah Satu, Aku Sayang Ibu

Now Playing:

Satu Tambah Satu oleh Aku Sayang Ibu

Masukkan Perintah: \pm

Masukkan Nama Lagu dan Penyanyi: Dua Dua, Juga Sayang Ayah

Now Playing:

Satu Tambah Satu oleh Aku Sayang Ibu

Dua Dua oleh Juga Sayang Ayah

Masukkan Perintah: \geq

Now Playing:

-> Satu Tambah Satu oleh Aku Sayang Ibu

Dua Dua oleh Juga Sayang Ayah

Masukkan Perintah: \geq

Now Playing:

Satu Tambah Satu oleh Aku Sayang Ibu

-> Dua Dua oleh Juga Sayang Ayah

Masukkan Perintah: \pm

Masukkan Nama Lagu dan Penyanyi: Tiga Tiga, Sayang Ayang

Now Playing:

Satu Tambah Satu oleh Aku Sayang Ibu

-> Dua Dua oleh Juga Sayang Ayah

Tiga Tiga oleh Sayang Ayang

Masukkan Perintah: \mathbb{R}

Now Playing:

-> Satu Tambah Satu oleh Aku Sayang Ibu

Dua Dua oleh Juga Sayang Ayah

Tiga Tiga oleh Sayang Ayang

Masukkan Perintah: \leq

Now Playing:

Satu Tambah Satu oleh Aku Sayang Ibu

Dua Dua oleh Juga Sayang Ayah

-> Tiga Tiga oleh Sayang Ayang

Masukkan Perintah: $=$

Now Playing:

Satu Tambah Satu oleh Aku Sayang Ibu

Dua Dua oleh Juga Sayang Ayah

Masukkan Perintah: \geq

Now Playing:

-> Satu Tambah Satu oleh Aku Sayang Ibu
Dua Dua oleh Juga Sayang Ayah

Masukkan Perintah: E

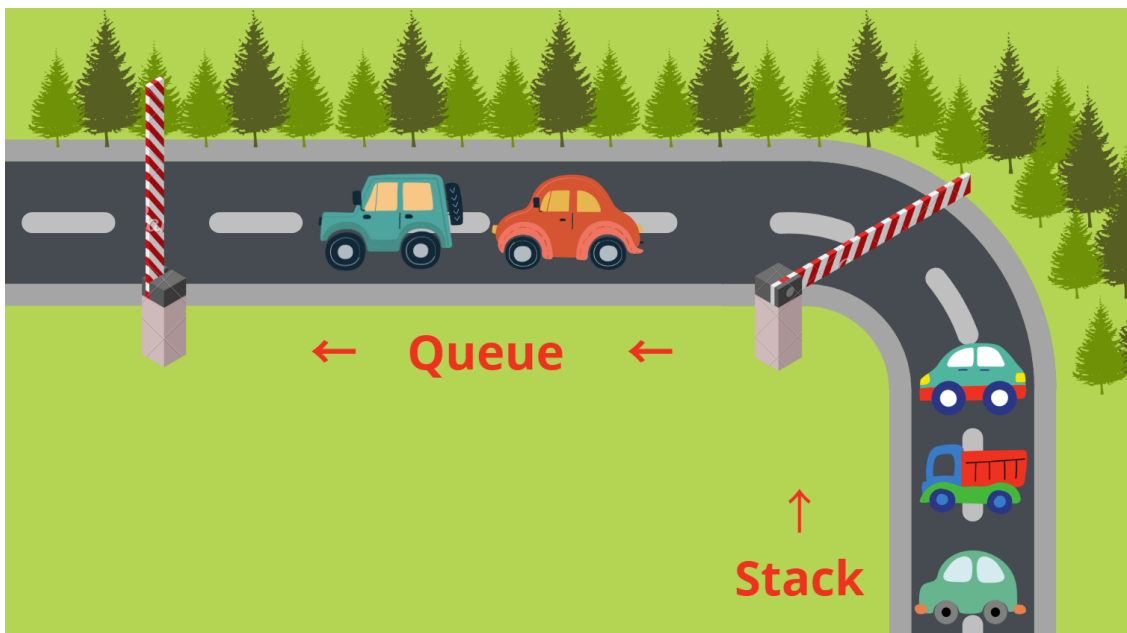
Soal 2

Suatu negara bernama Arstotzka adalah sebuah negara yang damai aman sentosa dan sejahtera. Karena kesejahteraan negaranya, banyak pengunjung ataupun imigran yang datang untuk bekerja, rekreasi, ataupun sekadar mengunjungi keluarga. Di Arstotzka, kamu adalah inspektur yang ditugaskan di gerbang perbatasan ini. Kedamaian yang ada membuat seluruh pengunjung diperbolehkan masuk sehingga kamu hanya perlu menghitung dan mendata pengunjung gerbang perbatasan ini.



Gambar 1 Gerbang perbatasan Arstotzka

Agar berjalan teratur, pendataan menggunakan dasar stack dan queue. Dasar stack digunakan dalam mengontrol mobil yang akan masuk setiap harinya, sementara queue digunakan dalam pos untuk proses monitoring mobil di dalam gerbang. Berikut ilustrasi beserta panduan dari buku manual Arstotzka.



Gambar 2 Alur imigrasi di gerbang Aristotzka

Panduan pendataan imigrasi

- Stack sudah berisi data imigran yang akan datang pada hari tersebut
- Queue (atau pos) kosong pada permulaan hari
- Pos memiliki dua gerbang:
 1. Gerbang 1, antara pos dan negara luar
 2. Gerbang 2, antara Arstotzka dan pos
- Inspektur memiliki lima perintah yang dapat dipakai
 1. Next : Membuka Gerbang 1
 2. Done : Membuka Gerbang 2
 3. Observe : Menampilkan data imigran dalam pos (antara Gerbang 1 & Gerbang 2)
 4. Monitor : Menampilkan jumlah imigran yang sudah memasuki Arstotzka (sudah melewati Gerbang 2)
 5. Rest : Menyelesaikan hari
- Inspektur diminta melaporkan feedback jika terjadi hal berikut:
 1. Hari libur : minggu, hari tidak valid
 2. Pengunjung habis : stack kosong saat Next
 3. Pos kosong : queue kosong saat Done

Buatlah program yang merepresentasikan alur imigrasi Arstotzka ini

Hint: Gunakan operasi Pop, Enqueue, isEmpty, dll di modul untuk memudahkan implementasi perintah

Contoh Eksekusi Program (garis bawah menandakan input)

```
#1
Masukkan hari: minggu.txt
Hari libur!

#2
Masukkan hari: senin.txt

>> Monitor
Jumlah imigran adalah 0

>> Next
>> Done
>> Monitor
Jumlah imigran adalah 1

>> Next
Pengunjung habis!
```



```

>> Rest
Selesai
#3
Masukkan hari: selasa.txt

>> Done
Pos kosong!

>> Next
>> Next
>> Observe
1 Jamie Adeyemi
2 Ferry Pier

>> Monitor
Jumlah imigran adalah 0

>> Rest
Selesai

#4
Masukkan hari: rabu.txt

>> Next
>> Next
>> Observe
1 Alyosha Egorov
2 Efia Danso

>> Next
>> Next
>> Next
>> Next
Pengunjung habis!

>> Observe
1 Alyosha Egorov
2 Efia Danso
3 Helsinki Pochinki
4 Kaela Kovalskia
5 Liam Byrne

>> Done
>> Done
>> Done
>> Observe
1 Kaela Kovalskia
2 Liam Byrne

>> Monitor

```

Jumlah imigran adalah 3

>> Rest
Selesai