

MIPS: Kantin PokPok

LAB - 0

Pengantar Organisasi Komputer
Semester Genap 2021/2022

Revisi 0

Selamat datang di lab Pengantar Organisasi Komputer semester genap tahun ajaran 2020/2021. Pada lab ini kita akan belajar dasar-dasar dari bahasa MIPS assembly.

Jangan lupa untuk selalu berkonsultasi dengan manual Guide to MIPS dan materi lab yang ada di scele, manual tersebut sangatlah berguna untuk kedepannya.

Selamat mengerjakan!

Judul Latihan	File yang Disubmit	Bobot Nilai
Latihan 1 : KANTIN POKPOK	latihan1.asm	60
Latihan 2 : KANTIN POKPOK JUSTIFIED	latihan2.asm	40
Latihan 3 : KANTIN POKPOK (BONUS)	latihanBonus.asm	10

Mohon kumpulkan semua file jawaban Anda (berupa file-file yang diminta pada kolom 'File yang Disubmit') dalam bentuk zip dengan format : **Lab0_KodeAsdos_Nama_NPM.zip**

Contoh: Lab0_RT_RicoTadjudin_1906398364.zip

Keterlambatan akan dikurangi 20 point jika ≤ 2 jam setelah deadline, 50 point ≤ 6 jam setelah deadline, dan tidak akan diterima setelahnya

Latihan 1 – Kantin PokPok (Max Score : 60)

Setelah berhasil “menaklukkan” pembelajaran mata kuliah pengantar organisasi komputer (POK) dengan sangat baik, Anda berpikir bahwa Anda dapat menaklukkan dunia dengan kemampuan yang Anda punya.

Sebelum dapat menaklukkan dunia, Anda terlebih dahulu harus mencari pengalaman dengan bekerja pada suatu proyek pekerjaan. Anda memiliki kemauan, tetapi tidak tahu harus melamar kemana.

Beberapa hari berlalu, Anda mendapatkan notifikasi lowongan kerja dari POKStreet mengenai peran sebagai programmer assembly pada suatu kantin bernama PokPok.

Kantin ini merupakan kantin yang unik karena **hanya menampung tiga menu utama** yaitu **Nasi Ayam Goreng, Nasi Udang Goreng, dan Nasi Ikan Goreng**. Sebagai manajer dari kantin ini Peokra berharap kamu dapat membantunya mengimplementasikan suatu sistem pemesanan sederhana di kantin ini. Tentunya gajimu akan sangat menjanjikan :D.

Format Input:

1 buah integer berupa nomor menu pesanan.

Input dijamin merupakan input yang valid sesuai dengan nomor menu yang tersedia.

Format Output:

Sebuah string dengan format “**Anda memesan: <Nama Pesanan>**”. Dengan nama pesanan yang bersesuaian dengan input integer yang diberikan.

Contoh eksekusi program:

```
Mars Messages Run I/O
Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan pesanan Anda (tuliskan dengan angka menu): 1
Anda memesan: Nasi Ayam Goreng
```

Petunjuk:

- "Masukkan pesanan Anda (tuliskan dengan angka menu): " adalah pesan untuk meminta input. Dapat dilakukan dengan print biasa pesan tersebut, lalu lakukan permintaan input berupa integer.
- Input yang diisi oleh user pada terminal "Run I/O" adalah integer jawaban jumlah.
Pada contoh di atas, user mengisi input berupa angka integer 1.
- Keluaran "Anda memesan: " dapat langsung ditaruh pada bagian data. Dalam artian lain hanya perlu di print oleh kodemu.
- Informasi nomor pesanan harus diambil dari input yang telah diminta (Tidak boleh langsung dari bagian data).
- Cara print string, print integer, meminta input integer ada di modul dan guide to mips yang telah diberikan sebagai materi awal Lab 0 ini.
- Jika ingin membuat new line pada string, bisa ditambahkan '\n' pada string.
- .asciiz adalah tipe data yang paling tepat jika Anda ingin membuat beberapa line terpisah, namun Anda juga bisa menggunakan .ascii biasa. Perbedaan dapat dilihat di guide to mips.
- **Jangan lupa melakukan syscall untuk exit di akhir!**

CATATAN :

Dokumentasikan lah program anda dengan lengkap dengan tanda # untuk comment. Akan ada penalti jika tidak ada dokumentasi nilai anda akan dikurangi 10 point.

Latihan 2 – Kantin PokPok Justified (Max Score : 40)

Senang dengan hasil kerja yang kamu berikan selama beberapa hari kamu melakukan masa *probation* di kantormu, Bosmu, Peokra menawarkanmu suatu proyek pembenaran dari proyek yang sudah kamu kerjakan sebelumnya. Dengan menyetujui dan menyelesaikan proyek ini, kamu akan mendapatkan promosi dan menaikkan gaji bulanan. Penawaran yang sangat menarik bukan ?!?!.

Dengan penawaran tersebut kamu mulai menanyakan requirement untuk dapat menyelesaikan proyek tersebut. Pada kenyataannya requirement tersebut tidaklah sulit. Kamu hanya diminta untuk melakukan validasi pesanan yang dimasukkan oleh user. Selama user memasukkan input yang tidak valid **kamu diminta untuk dapat melakukan teknik yang dapat dikatakan mirip seperti error handling** hingga user memasukkan input yang valid.

Format Input:

1 buah integer berupa nomor menu pesanan.

Format Output:

Sebuah string dengan format “**Anda memesan: <Nama Pesanan>**” tanpa tanda petik. Dengan nama pesanan yang bersesuaian dengan input integer yang diberikan. Pada kasus input tidak valid, akan dihasilkan sebuah string “**Masukkan Anda tidak valid**” tanpa tanda petik.

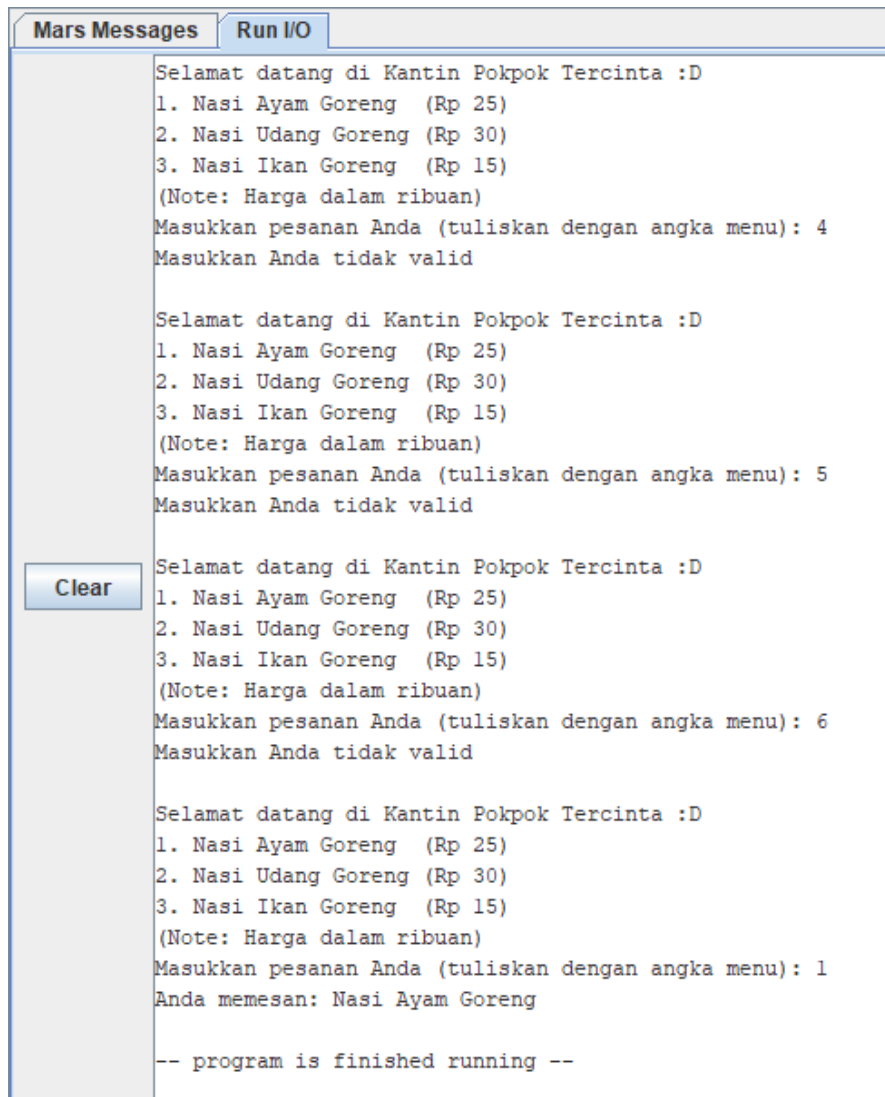
Contoh eksekusi program:

```
Mars Messages Run I/O
Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan pesanan Anda (tuliskan dengan angka menu): 4
Masukkan Anda tidak valid

Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan pesanan Anda (tuliskan dengan angka menu): |
```

Catatan:

4 merupakan nomor pesanan yang tidak valid sehingga program tetap meminta input hingga dimasukkan input yang valid.



```
Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan pesanan Anda (tuliskan dengan angka menu): 4
Masukkan Anda tidak valid

Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan pesanan Anda (tuliskan dengan angka menu): 5
Masukkan Anda tidak valid

Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan pesanan Anda (tuliskan dengan angka menu): 6
Masukkan Anda tidak valid

Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan pesanan Anda (tuliskan dengan angka menu): 1
Anda memesan: Nasi Ayam Goreng

-- program is finished running --
```

Catatan:

4, 5, 6 bukanlah nomor pesanan yang valid, sehingga program akan terus meminta masukan hingga dimasukkan suatu input yang valid pada kasus ini angka 1 merupakan input yang valid sehingga program pun dapat diselesaikan.

CATATAN :

Dokumentasikan lah program anda dengan lengkap dengan tanda # untuk comment. Akan ada penalti jika tidak ada dokumentasi nilai anda akan dikurangi 10 point.

Latihan Bonus – Kantin PokPok Bonus (Max Score : 10)

Puas dengan hasil kerja yang dapat kamu berikan, Peokra ingin menantangmu untuk terakhir kalinya sebelum kamu diberikan suatu *privilege* yang sangat besar di kantormu.

Mengingat baiknya keberadaan sistem yang sudah ada, Peokra masih memikirkan suatu fitur lagi. Selama ini banyak user yang komplain akan kekurangan sistem tersebut dikarenakan dapat memesan hanya satu pesanan suatu waktu.

Oleh karena itu, dia ingin suatu sistem yang **dapat memesan banyak pesanan** dalam suatu waktu. Sistem yang dia inginkan juga unik karena sistem ini **diharapkan dapat mencatat berapa kali user memasukkan pesanan yang tidak valid**. Selain itu, fitur yang tidak kalah penting adalah **perhitungan total harga dari semua pesanan yang ada**. Dengan ini, Peokra berharap perusahaannya memiliki sistem kantin terbaik sejagat raya.

Format Input:

1 buah integer N berupa jumlah pesanan.

N buah baris masukkan yang menandakan nomor menu pesanan.

Format Output:

Sebuah string dengan format “**Anda memesan: <Nama Pesanan>**” tanpa tanda petik pada setiap baris setiap kali user selesai melakukan input nomor menu yang diinginkan. Dengan nama pesanan yang bersesuaian dengan input integer yang diberikan.

Pada setiap kasus input yang tidak valid, akan dihasilkan sebuah string “**Masukkan Anda tidak valid**” tanpa tanda petik.

Pada akhir program, akan dituliskan sebuah string dengan format **"Terdapat <jumlah_pesanan_tidak_valid> pesanan tidak valid"** tanpa tanda petik.

Kemudian akan diikuti oleh string dengan format **"Total harga pesanan Anda (Dalam ribuan): <total_harga>"** tanpa tanda petik pada baris setelahnya.

Contoh eksekusi program:

Mars Messages	Run I/O
<pre>Selamat datang di Kantin Pokpok Tercinta :D 1. Nasi Ayam Goreng (Rp 25) 2. Nasi Udang Goreng (Rp 30) 3. Nasi Ikan Goreng (Rp 15) (Note: Harga dalam ribuan) Masukkan jumlah pesanan: 3 Daftar pesanan (Tuliskan dengan angka menu): 1 Anda memesan: Nasi Ayam Goreng 4 Masukkan Anda tidak valid 1 Anda memesan: Nasi Ayam Goreng Terdapat 1 pesanan tidak valid Total harga pesanan Anda (Dalam ribuan): 50 -- program is finished running --</pre>	

Catatan:

- Total harga yang dicatat adalah 50 karena terdapat dua menu Nasi Ayam Goreng yang dicatat
 $(25 + 25) = 50$.
- Terdapat satu pesanan tidak valid yaitu ketika user memasukkan angka "4" sebagai input.


```
Mars Messages Run I/O
Selamat datang di Kantin Pokpok Tercinta :D
1. Nasi Ayam Goreng (Rp 25)
2. Nasi Udang Goreng (Rp 30)
3. Nasi Ikan Goreng (Rp 15)
(Note: Harga dalam ribuan)
Masukkan jumlah pesanan: 5
Daftar pesanan (Tuliskan dengan angka menu):
1
Anda memesan: Nasi Ayam Goreng
5
Masukkan Anda tidak valid
6
Masukkan Anda tidak valid
3
Anda memesan: Nasi Ikan Goreng
4
Masukkan Anda tidak valid
Terdapat 3 pesanan tidak valid
Total harga pesanan Anda (Dalam ribuan): 40
-- program is finished running --
```

Catatan:

- Total harga yang dicatat adalah 40 karena terdapat menu Nasi Ayam Goreng dan Nasi Ikan Goreng yang dicatat.
 $(25 + 15) = 40$.
- Terdapat tiga pesanan tidak valid yaitu ketika user memasukkan angka "5", "6", dan "4" sebagai input.

CATATAN :

Dokumentasikan lah program anda dengan lengkap dengan tanda # untuk comment. Akan ada penalti jika tidak ada dokumentasi nilai anda akan dikurangi 10 point.