
Started on Thursday, 7 October 2021, 8:24 PM

State Finished

Completed on Thursday, 7 October 2021, 8:26 PM

Time taken 1 min 32 secs

Marks 170.00/180.00

Grade **94.44** out of 100.00

Time limit	1 s
Memory limit	64 MB

Dengan mengadaptasi Word Machine yang telah Anda kerjakan sebagai tugas pra-praktikum, buatlah modifikasi terhadap Word Machine tersebut sehingga menjadi **Token Machine** yang memiliki spesifikasi sebagai berikut:

- Token machine membaca pita karakter yang hanya berisi rangkaian bilangan bulat dan operator. Ada lima operator yang dikenali, yaitu: +, -, *, /, dan ^.
- Token machine akan membaca pita karakter dan mengembalikan nilai currentToken yang bertipe Token, dengan struktur sebagai berikut:

```
type Token : < tkn : character, {token yang dibaca}
              val : integer {nilai bilangan apabila tkn = 'b'}
            >
```

Keterangan tipe Token adalah sebagai berikut:

- tkn berisi token yang diakuisisi. Ada 6 nilai yang mungkin bagi tkn yaitu '+', '-', '*', '/', dan '^' masing-masing untuk setiap operator yang terdefinisi, serta 'b' untuk token yang merupakan bilangan.
- val berisi nilai dari bilangan yang diakuisisi, apabila tkn berisi 'b'.
- Apabila tkn adalah salah satu dari lima operator, maka nilai val diabaikan (diisi -1).
- Diasumsikan setiap token dalam pita karakter selalu dipisahkan spasi. Karena tidak mengenal tanda, maka semua bilangan yang diakuisisi pasti merupakan bilangan positif. Petunjuk untuk proses konversi menjadi bilangan:

```
321 = (((3 * 10) + 2) * 10) + 1
```

Header file ADT Token dapat dilihat pada file: [tokenmachine.h](#).

Submit **tokenmachine.c**.

Contoh isi pita karakter dan keluaran token machine setiap akuisisi:

Masukan (isi pita karakter)	Token yang dibaca
25000 + 300 .	<'b',25000> <'+',-1> <'b',300>
34 * 4 + 12 / 2 ^ 3.	<'b',34> <('*'),-1> <'b',4> <'+',-1> <'b',12> <('/',-1> <'b',2> <('^'),-1> <'b',3>

C

 [tokenmachine.c](#)

Score: 70

Blackbox

Score: 70

Verdict: Wrong answer

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.65 MB
2	10	Accepted	0.00 sec, 1.55 MB
3	10	Accepted	0.00 sec, 1.58 MB
4	10	Accepted	0.00 sec, 1.64 MB

No	Score	Verdict	Description
			0.00 sec, 1.58 MB

6	10	Accepted	0.00 sec, 1.57 MB
7	10	Accepted	0.00 sec, 1.55 MB
8	0	Wrong answer	0.00 sec, 1.55 MB

Question **2**

Correct

Mark 100.00 out
of 100.00

Time limit

1 s

Memory limit

64 MB

Enkripsi adalah proses mengamankan suatu informasi dengan membuat informasi tersebut tidak dapat dibaca tanpa bantuan pengetahuan khusus. Caesar Cipher (sandi Caesar) adalah salah satu teknik enkripsi. Cara kerja dari Caesar Cipher adalah dengan cara setiap huruf pada teks terang (plaintext) digantikan oleh huruf lain yang memiliki selisih posisi tertentu dalam alfabet. Misal jika terdapat sebuah kata IBU dan dilakukan penggeseran sebanyak 10, maka hasil cipher adalah SLE.

Buatlah sebuah program utama (**file: cipher.c**) yang menggunakan word machine dan membaca sebuah pita karakter, lalu menerjemahkan seluruh kata dalam pita karakter tersebut sesuai dengan aturan Caesar Cipher dan mencetaknya ke layar. Kemudian, akhiri dengan titik dan newline.

Banyaknya penggeseran diambil dari panjang kata pertama, dan pita karakter tidak mungkin kosong. Petunjuk : Kode ASCII untuk A adalah 65 dan untuk Z adalah 90.

Contoh masukan dan keluaran:

Masukan (isi pita karakter)	Output (Ke Layar)	Keterangan
ANI PERGI KE PASAR DENGAN SEPEDA.	DQL SHUJL NH SDVDU GHQJDQ VHSHGD.	Banyaknya penggeseran = 3
KEBAKARAN HUTAN ITU MELUAS HINGGA ASAP KABUT YANG DITIMBULKAN BERDAMPAK HINGGA SINGAPURA DAN MALAYSIA.	TNKJTJAJW QDCJW RCD VNUDJB QRWPPJ JBJY TJKDC HJWP MRCRVKDUTJW KNAMJVYJT QRWPPJ BRWPJYDAJ MJW VJUJHBRJ.	Banyaknya penggeseran = 9

C

 [cipher.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	13	Accepted	0.00 sec, 1.59 MB
2	13	Accepted	0.00 sec, 1.58 MB
3	13	Accepted	0.00 sec, 1.51 MB
4	13	Accepted	0.00 sec, 1.45 MB
5	12	Accepted	0.00 sec, 1.58 MB
6	12	Accepted	0.00 sec, 1.54 MB
7	12	Accepted	0.00 sec, 1.47 MB
8	12	Accepted	0.00 sec, 1.52 MB

[◀ tokenmachine.h](#)

Jump to...

[queue.h ▶](#)

