

Tugas Personal ke-1

Week 2

Soal.

1. Jelaskan peran compiler dalam proses penerjemahan bahasa pemrograman. Apa perbedaan hasil akhir dari proses kompilasi dibandingkan dengan proses interpretasi? (LO 1 : 30%)
2. Bahasa pemrograman C tidak memiliki tipe boolean. Jelaskan bagaimana *C compiler* dapat menerjemahkan *if-statement* menjadi *three-address code*! (LO 1 : 30%)
3. Buatlah DAG untuk ekspresi berikut: (LO 2 : 40%)

$$(a + b) * (c + d) + (a + b) * (e - f) - (c + d)$$

Jawaban:

1. Peran Compiler dalam proses penerjemahan Bahasa pemrograman, apa perbedaan hasil akhir dari proses kompilasi dibandingkan dengan proses interpretasi

Compiler dalam proses penerjemahan bahasa pemrograman untuk menerjemahkan seluruh kode sumber yang ditulis dalam bahasa pemrograman tingkat tinggi menjadi bahasa mesin yang dapat dipahami dan dijalankan oleh komputer.

Adapun struktur/peran compiler dalam proses penerjemahan:

a. Lexical Analysis

compiler membaca kode sumber karakter demi karakter dan mengelompokkannya menjadi unit-unit yang lebih kecil yang disebut token.

b. Syntax Analysis

compiler memeriksa apakah susunan token tersebut sesuai dengan aturan tata bahasa dari bahasa pemrograman.

c. Semantic Analysis

Penganalisis semantik menggunakan pohon sintaksis dan informasi dalam tabel simbol untuk memeriksa program sumber agar sesuai dengan konsistensi semantik dalam definisi bahasa.

d. Intermediate Code Generation

compiler akan menghasilkan kode perantara (intermediate code). Kode ini merupakan bentuk representasi program yang berada di antara bahasa tingkat tinggi dan bahasa mesin.

e. Code Optimization

proses memperbaiki kode perantara agar program menjadi lebih efisien tanpa mengubah hasil program.

f. Code generator

menerjemahkan kode perantara menjadi kode mesin atau assembly yang dapat dijalankan oleh prosesor.

g. Symbol-Table Management

Tabel simbol adalah struktur data yang berisi catatan untuk setiap nama variabel, dengan bidang untuk atribut nama. Struktur data harus didesain agar kompiler dapat menemukan record untuk setiap nama dengan cepat dan menyimpan atau mengambil data dari record tersebut dengan cepat.

Perbedaan hasil akhir kompilasi dan interpretasi

Kompilasi

- Seluruh program diterjemahkan sekaligus sebelum dijalankan.
- Hasil dari proses kompilasi adalah file executable atau kode mesin.
- Setelah program dikompilasi, program dapat dijalankan tanpa membutuhkan compiler lagi.

Interpretasi

- Program diterjemahkan baris demi baris saat program dijalankan.
- Tidak menghasilkan file executable permanen.
- Program membutuhkan interpreter setiap kali dijalankan.

2. bagaimana *C compiler* dapat menerjemahkan *if-statement* menjadi *three-address code* bahasa C tidak memiliki tipe data boolean eksplisit, *compiler* menggunakan nilai integer untuk menentukan aliran program (0 untuk False dan selain 0 untuk True). Proses penerjemahannya mengikuti langkah-langkah berikut:

a. Representasi Kondisi:

Compiler mengevaluasi ekspresi di dalam *if*. Jika ekspresi tersebut adalah operasi relasional, compiler akan menghasilkan instruksi lompatan bersyarat. Jika ekspresi adalah variabel tunggal, compiler membandingkan nilainya dengan angka 0.

b. Penggunaan Label dan Jumps

three-address code menggunakan Label sebagai penanda alamat memori dan instruksi goto untuk mengontrol aliran eksekusi. Struktur dasar untuk *if (E) S1 else S2* adalah:

- Jika E bernilai salah (sama dengan 0), program melompat ke blok *else* atau akhir pernyataan.
- Jika E bernilai benar (tidak sama dengan 0), program mengeksekusi S1 lalu melompat ke akhir pernyataan.

c. Struktur Penulisan 3AC

contoh:

$\text{if } (a < b)$ $x = y + z;$

compiler harus:

- Mengevaluasi kondisi $a < b$
- Jika benar: jalankan $x = y + z$
- Jika salah: lompat ke instruksi berikutnya

Compiler akan menghasilkan bentuk three-address code seperti berikut:

```
if a < b goto L1
goto L2
L1: x = y + z
L2: # logic selanjutnya
```

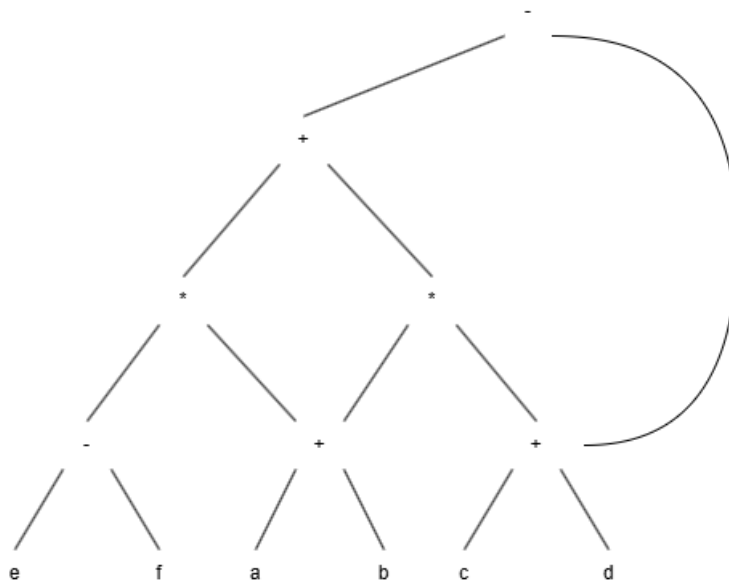
yang berarti:

- Jika kondisi benar, program lompat ke label **L1**.
- jika kondisi salah, Program melewati blok if.
- Jika kondisi benar, perintah L1 dijalankan.
- Program melanjutkan ke instruksi L2.

3. DAG dari soal yang diberikan:

$(a + b) * (c + d) + (a + b) * (e - f) - (c + d)$

berdasarkan contoh DAG yang diberikan pada LN05, bisa dibuat seperti ini



Dengan urutan seperti berikut:

- a. $t1 = e - f$
- b. $t2 = a + b$
- c. $t3 = c + d$
- d. $t4 = t1 * t2$
- e. $t5 = t2 * 3$
- f. $t6 = t4 * t5$
- g. $t7 = t6 - t3$