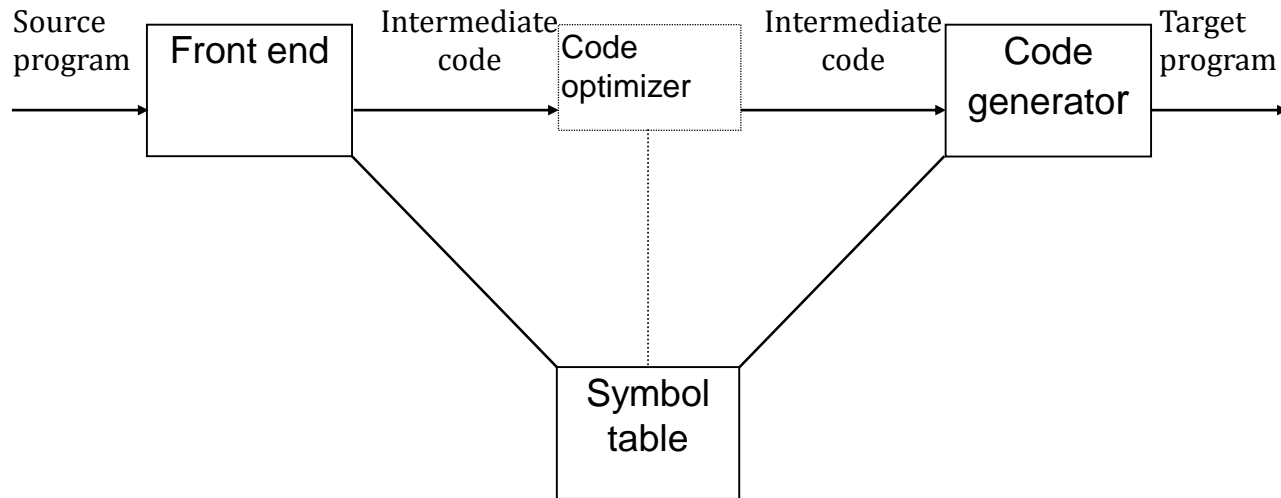


CODE GENERATION

CODE GENERATION

- Merupakan tahap terakhir dari kompiler. Input code generator adalah representasi intermediate dari source program, sedangkan outputnya adalah target program yang ekuivalen.

Skema Code Generator



Syarat Code Generator

- Output code harus benar
- Output code berkualitas tinggi yaitu:
 - harus menggunakan resources dari target machine secara efektif
- Output code harus efisien

Masalah Code Generator

- Input untuk code generator
- Target program
- Memory Management
- Instruction Selection
- Register Allocation
- Pemilihan Urutan Evaluation
- Pendekatan Code Generator

Bentuk dari target program

- Absolute Machine Language
 - mempunyai keuntungan dapat ditempatkan dalam lokasi tetap di memory dan dapat dieksekusi langsung
- Relocatable Machine Language
 - memungkinkan subprogram dikompilasi secara terpisah (flexibility).
 - Satu set object module dapat link / load bersama-sama pada eksekusi linking-loader (perlu usaha ekstra)
- Assembly Language
 - keuntungannya adalah adanya kemudahan proses code generation untuk menggenerate symbolic instruction dan menggunakan fasilitas makro dari assembler untuk menggenerate code

Contoh

Three-address statement

Code

- $x := y + z$
- MOV y, R0 /* load y ke register R0 */
- ADD z, R0 /* add z ke R0 */
- MOV R0, x /* store R0 ke dalam x */

Redudant

Three address

- $a := b + c$
- $d := a + e$

Code

- MOV b, R0
- ADD c, R0
- MOV R0, a
- MOV a, R0
- ADD e, R0
- MOV R0, d

TERIMA KASIH