

# Bab 17

## Evaluasi dan Refleksi Kompetensi

### Sub-CPMK yang Dicakup dalam Bab Ini:

- **Sub-CPMK 1.1-6.2:** Mengevaluasi seluruh capaian pembelajaran mata kuliah Teknik Kompilasi

### 17.1 Latihan Mandiri Komprehensif

Bagian ini menyajikan kumpulan soal latihan untuk menguji pemahaman mahasiswa terhadap seluruh fase kompilasi yang telah dipelajari.

#### 17.1.1 Analisis Leksikal dan Sintaksis

Identifikasi token dan gambarkan *parse tree* untuk ekspresi berikut:  $x = (10 + y) * 2;$ .

#### 17.1.2 Semantik dan Tabel Simbol

Gambarkan hierarki tabel simbol untuk kode program yang memiliki fungsi di dalam lingkup global dan variabel lokal dengan nama yang sama (*shadowing*).

#### 17.1.3 Generasi IR dan Optimasi

Tuliskan *Three-Address Code* (TAC) untuk struktur `while` loop dan lakukan *Constant Folding* jika terdapat ekspresi statis.

## 17.2 Evaluasi Capaian Pembelajaran Mata Kuliah

### 17.2.1 CPMK-1: Arsitektur Kompilator

Sub-CPMK	Teori	Praktik	Proyek	Nilai
1.1: Fase kompilasi	20%	15%	10%	45%
1.2: Struktur kompilator	20%	15%	10%	45%
1.3: Intermediate code	20%	15%	10%	45%

Tabel 17.1: Komponen Penilaian CPMK-1

### 17.2.2 CPMK-2: Lexer dan Parser

Sub-CPMK	Teori	Praktik	Proyek	Nilai
2.1: Regular expression	15%	25%	15%	55%
2.2: Finite automata	15%	25%	15%	55%
2.3: Parsing techniques	20%	20%	15%	55%
2.4: Parser generators	15%	25%	15%	55%

Tabel 17.2: Komponen Penilaian CPMK-2

### 17.2.3 CPMK-3: Semantic Analysis

Sub-CPMK	Teori	Praktik	Proyek	Nilai
3.1: Symbol table	15%	25%	15%	55%
3.2: Scope management	15%	25%	15%	55%
3.3: Type checking	20%	20%	15%	55%

Tabel 17.3: Komponen Penilaian CPMK-3

### 17.2.4 CPMK-4: Intermediate Code dan Optimasi

### 17.2.5 CPMK-5: Code Generation dan Runtime

### 17.2.6 CPMK-6: Tools dan Evaluasi

## 17.3 Refleksi Pembelajaran

### 17.3.1 Pertanyaan Reflektif

#### Bagian 1: Pemahaman Konsep

1. Konsep kompilator mana yang paling menantang bagi Anda? Mengapa?

Sub-CPMK	Teori	Praktik	Proyek	Nilai
4.1: Three-address code	15%	25%	15%	55%
4.2: Basic blocks	15%	25%	15%	55%
4.3: Local optimization	20%	20%	15%	55%

Tabel 17.4: Komponen Penilaian CPMK-4

Sub-CPMK	Teori	Praktik	Proyek	Nilai
5.1: Runtime environment	15%	25%	15%	55%
5.2: Memory layout	15%	25%	15%	55%
5.3: Code generation	20%	20%	15%	55%

Tabel 17.5: Komponen Penilaian CPMK-5

2. Bagaimana pemahaman Anda tentang kompilator berkembang selama semester ini?
3. Konsep mana yang paling relevan dengan pengembangan software modern?

### Bagian 2: Keterampilan Praktis

1. Implementasi mana yang memberikan insight terbesar tentang kompilasi?
2. Kesulitan teknis apa yang Anda hadapi dan bagaimana mengatasinya?
3. Keterampilan mana yang paling berharga untuk karir Anda?

### Bagian 3: Pengembangan Diri

1. Bagaimana mata kuliah ini mengubah cara Anda berpikir tentang programming?
2. Topik mana yang ingin Anda pelajari lebih dalam?
3. Bagaimana Anda akan menerapkan pengetahuan ini di proyek masa depan?

## 17.3.2 Self-Assessment Checklist

## 17.4 Portofolio Proyek

### 17.4.1 Struktur Portofolio

Portofolio proyek kompilator harus mencakup:

1. **Deskripsi Proyek:** Tujuan, scope, dan requirements
2. **Arsitektur:** Desain komponen dan interaksi
3. **Implementasi:** Code snippets dan penjelasan
4. **Testing:** Test cases dan hasil

Sub-CPMK	Teori	Praktik	Proyek	Nilai
6.1: Compiler tools	15%	25%	15%	55%
6.2: Performance evaluation	15%	25%	15%	55%

Tabel 17.6: Komponen Penilaian CPMK-6

Kompetensi	Belum	Sedang	Menguasai
Memahami fase kompilasi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Mengimplementasikan lexer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Membangun parser	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Melakukan semantic analysis	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Menghasilkan intermediate code	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Melakukan optimasi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Menghasilkan target code	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Menggunakan compiler tools	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Tabel 17.7: Self-Assessment Kompetensi

5. **Evaluasi:** Performance analysis dan benchmarking

6. **Refleksi:** Pembelajaran dan improvement

## 17.4.2 Kriteria Penilaian Portofolio

Aspek	Bobot
Koreksi dan kelengkapan	20%
Kualitas implementasi	25%
Pengujian dan validasi	20%
Dokumentasi	15%
Presentasi	10%
Refleksi dan pembelajaran	10%

Tabel 17.8: Kriteria Penilaian Portofolio

## 17.5 Feedback dan Continuous Improvement

### 17.5.1 Mekanisme Feedback

- **Formative Assessment:** Feedback selama pembelajaran
- **Peer Review:** Feedback dari teman sekelas
- **Self-Reflection:** Evaluasi diri
- **Instructor Feedback:** Feedback dari dosen

### 17.5.2 Action Plan for Improvement

1. **Identifikasi Gap:** Area yang perlu improvement
2. **Set Goals:** Target pencapaian yang spesifik
3. **Action Steps:** Langkah konkret untuk mencapai goals
4. **Timeline:** Jadwal implementasi
5. **Evaluation:** Cara mengukur progress

## 17.6 Kesimpulan dan Rekomendasi

### 17.6.1 Pencapaian Pembelajaran

Setelah menyelesaikan mata kuliah Teknik Kompilasi, mahasiswa diharapkan:

- Memahami prinsip-prinsip fundamental kompilasi
- Mampu mengimplementasikan compiler components
- Dapat menggunakan tools modern untuk pengembangan compiler
- Memiliki kemampuan analisis dan evaluasi performance
- Siap untuk pengembangan software yang lebih kompleks

### 17.6.2 Rekomendasi Lanjutan

- **Study Lanjut:** Advanced compiler design, optimization techniques
- **Aplikasi Praktis:** Bahasa programming, system programming
- **Research:** Compiler optimization, language design
- **Industry:** Compiler development, tool development

### 17.6.3 Final Reflection

Teknik Kompilasi adalah mata kuliah fundamental yang menghubungkan teori computer science dengan implementasi praktis. Pemahaman yang baik tentang kompilator memberikan fondasi kuat untuk pengembangan software yang efisien dan optimal.

*"The best way to learn compiler construction is to build a compiler."* - Andrew Appel

## Aktivitas Pembelajaran

1. **Self-Assessment:** Isi checklist kompetensi mandiri.
2. **Portfolio Construction:** Susun portofolio proyek semester.
3. **Reflective Essay:** Tulis refleksi pembelajaran satu semester.
4. **Peer Review:** Evaluasi portofolio rekan sekelas.
5. **Action Planning:** Buat rencana pengembangan diri pasca mata kuliah.

### Latihan dan Refleksi

1. Identifikasi Sub-CPMK yang paling sulit dicapai!
2. Susun bukti-bukti pencapaian kompetensi dalam portofolio!
3. Analisis kaitan antar modul dalam pengembangan compiler!
4. Evaluasi perkembangan keterampilan programming Anda!
5. Rencanakan langkah belajar selanjutnya di bidang software engineering!
6. **Refleksi:** Apa insight terbesar yang Anda dapatkan dari mata kuliah ini?

### Asesmen (Evaluasi Kinerja)

#### Instrumen Penilaian Akhir Semester

##### A. Evaluasi Portofolio

1. Kelengkapan komponen compiler (Lexer, Parser, Semantic, dsb)
2. Kualitas dokumentasi dan analisis
3. Hasil pengujian dan benchmarking

##### B. Presentasi Proyek

1. Demonstrasi fungsionalitas compiler
2. Penjelasan arsitektur dan design decisions
3. Tanya jawab teknis

**Rubrik Penilaian:** Lihat Lampiran A dan B

### Checklist Pencapaian Kompetensi

*Centang item berikut setelah Anda yakin telah menguasainya:*

- ☐ Saya memahami seluruh fase kompilasi dan interaksinya
- ☐ Saya dapat mengimplementasikan front-end dan back-end compiler
- ☐ Saya dapat melakukan optimasi dan performance evaluation
- ☐ Saya mahir menggunakan compiler tools modern
- ☐ Saya dapat mendokumentasikan proyek pengembangan compiler dengan baik
- ☐ Saya siap menerapkan prinsip kompilasi dalam rekayasa perangkat lunak

### Rangkuman

Bab ini merangkum seluruh perjalanan pembelajaran Teknik Kompilasi, melalui evaluasi CPMK, refleksi diri, dan penyusunan portofolio. Mahasiswa belajar mengevaluasi pencapaian kompetensi secara komprehensif.

#### Poin Kunci:

- Evaluasi kompetensi mencakup teori dan praktik implementasi
- Refleksi membantu internalisasi pemahaman konsep
- Portofolio proyek menjadi bukti nyata pencapaian Sub-CPMK
- Continuous improvement penting untuk pengembangan karir profesional
- Mata kuliah ini memberikan fondasi kuat dalam computer science

**Kata Kunci:** *Evaluasi Kompetensi, Refleksi, Portofolio, CPMK, Self-Assessment, Continuous Improvement*