



LABORATORIUM  
TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS AHMAD DAHLAN



# PETUNJUK PRAKTIKUM

EDISI KURIKULUM OBE

DASAR PEMROGRAMAN

Penyusun:

Bambang Robi'in, S.T., M.T.

Dewi Pramudi Ismi, S.T., M. CompSc.

Fitri Indra Indikawati, S. Kom., M. Eng.

Supriyanto, S.T., M.T.

Guntur Maulana Zamroni, B.Sc., M. Kom.

2021

## HAK CIPTA

### PETUNJUK PRAKTIKUM DASAR PEMROGRAMAN

**Copyright© 2021,**

Bambang Robi'in, S.T., M.T.

Dewi Pramudi Ismi, S.T., M. CompSc.

Fitri Indra Indikawati, S. Kom., M. Eng.

Supriyanto, S.T., M.T.

Guntur Maulana Zamroni, B.Sc., M. Kom.

**Hak Cipta dilindungi Undang-Undang**

Dilarang mengutip, memperbanyak atau mengedarkan isi buku ini, baik sebagian maupun seluruhnya, dalam bentuk apapun, tanpa izin tertulis dari pemilik hak cipta dan penerbit.

**Diterbitkan oleh:**

**Program Studi Teknik Informatika**

Fakultas Teknologi Industri

Universitas Ahmad Dahlan

Jalan Ring Road Selatan, Tamanan, Banguntapan, Bantul Yogyakarta 55166

**Penulis**

: Bambang Robi'in, S.T., M.T.

Dewi Pramudi Ismi, S.T., M. CompSc.

Fitri Indra Indikawati, S. Kom., M. Eng.

Supriyanto, S.T., M.T.

Guntur Maulana Zamroni, B.Sc., M. Kom.

**Editor**

: Laboratorium Teknik Informatika, Universitas Ahmad Dahlan

**Desain sampul**

: Laboratorium Teknik Informatika, Universitas Ahmad Dahlan

**Tata letak**

: Laboratorium Teknik Informatika, Universitas Ahmad Dahlan

**Ukuran/Halaman**

: 21 x 29,7 cm / 116 halaman

**Didistribusikan oleh:**



**Laboratorium Teknik Informatika**

Universitas Ahmad Dahlan

Jalan Ring Road Selatan, Tamanan, Banguntapan, Bantul Yogyakarta 55166

Indonesia

## KATA PENGANTAR

Alhamdulillah, Segala Puji bagi Allah swt yang telah melimpahkan Rahmat dan KaruniaNya sehingga kami tim dosen pengampu mata kuliah Dasar-Dasar Pemrograman dapat menyelesaikan penyusunan Buku Petunjuk Praktikum ini. Dasar-dasar Pemrograman merupakan mata kuliah yang menekankan pada aspek pengetahuan sekaligus aspek ketrampilan. Aspek ketrampilan dibangun melalui kegiatan-kegiatan praktik yang dilaksanakan di kelas maupun di laboratorium. Buku petunjuk praktikum ini merupakan panduan bagi mahasiswa dalam melaksanakan pembelajaran Dasar-Dasar Pemrograman di laboratorium. Sebagai penyusun kami menyadari bahwa masih banyak terdapat kekurangan dalam buku petunjuk praktikum ini, oleh karena itu kritik dan saran yang membangun sangat kami harapkan. Akhir kata, kami berharap agar buku petunjuk praktikum ini dapat bermanfaat bagi mahasiswa dalam mata kuliah Dasar-dasar Pemrograman.

Yogyakarta, 1 Agustus 2021

Tim Penyusun

## DAFTAR PENYUSUN

### **Bambang Robi'in, S.T., M.T.**



Penulis adalah Dosen program studi Teknik Informatika UAD dengan bidang keahlian Multimedia dan Mobile Computing. Pendidikan S1 Teknik Informatika UAD dan Pendidikan S2 Teknik Informatika UAJY.

### **Dewi Pramudi Ismi, S.T., M. CompSc.**



Penulis adalah Dosen program studi Teknik Informatika UAD dengan bidang keahlian Machine Learning. Pendidikan S1 Teknik Informatika ITB, Pendidikan S2 Computer Science University of Queensland- Australia, dan saat ini sedang menempuh pendidikan S3 Ilmu Komputer UGM.

### **Fitri Indra Indikawati, S. Kom., M. Eng.**



Penulis adalah Dosen program studi Teknik Informatika UAD dengan bidang keahlian Big Data Analytics, IoT Data Management. Pendidikan S1 Teknik Informatika ITS dan Pendidikan S2 Big Data Pusan National University.

### **Supriyanto, S.T., M.T.**



Penulis adalah Dosen program studi Teknik Informatika UAD dengan bidang keahlian Media Digital dan Game. Pendidikan S1 Teknik Informatika UAD dan Pendidikan S2 Teknik Elektro ITB.

### **Guntur Maulana Zamroni, B.Sc., M. Kom.**



Penulis adalah Dosen program studi Teknik Informatika UAD dengan bidang keahlian Mobile Forensic dan Software Engineering. Pendidikan S1 Staffordshire University – Inggris dan Pendidikan S2 Teknik Informatika UAD.

## HALAMAN REVISI

Yang bertanda tangan di bawah ini:

Nama : Bambang Robi'in, S.T., M.T.

NIP/NIY : 197907202005011002

Jabatan : Dosen Pengampu Mata Kuliah **Dasar Pemrograman**

Dengan ini menyatakan pelaksanaan Revisi Petunjuk Praktikum **Dasar Pemrograman** untuk Program Studi Teknik Informatika telah dilaksanakan dengan penjelasan sebagai berikut:

No	Keterangan Revisi	Tanggal Revisi	Nomor Modul
1	a. Penambahan latihan pada setiap pertemuan. b. Penambahan modul praktikum pointer c. Penambahan modul library buatan	25 Juli 2019	PP/018/I/R1
2	a. Perubahan format modul OBE b. Penambahan Capaian Pembelajaran c. Perubahan total aloaksi waktu d. Perubahan 13 kali pertemuan praktikum menjadi 14 kali pertemuan praktikum e. Penggabung praktikum 9 library dengan praktikum 8 fungsi f. Penambah praktikum 9 fungsi rekursif. g. Penambahan praktikum 11 Array Dua Dimensi	5 Agustus 2021	PP/018/I/R2

Yogyakarta, 1 Agustus 2021

Penyusun



**Bambang Robi'in, S.T., M.T.**

**NIP. 197907202005011002**

## HALAMAN PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : Lisna Zahrotun, S.T., M.Cs.

NIK/NIY : 60150773

Jabatan : Kepala Laboratorium Teknik Informatika

Menerangkan dengan sesungguhnya bahwa Petunjuk Praktikum ini telah direview dan akan digunakan untuk pelaksanaan praktikum di Semester Gasal Tahun Akademik 2021/2022 di Laboratorium Praktikum Teknik Informatika, Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Ahmad Dahlan.

Yogyakarta, 1 Agustus 2021

Mengetahui,  
Ketua Kelompok Keilmuan Relata



**Guntur Maulana Zamroni, B.Sc. M. Kom**  
NIY. 60181172

Kepala Laboratorium Teknik Informatika



**Lisna Zahrotun, S.T., M.Cs.**  
NIY. 60150773

## VISI DAN MISI PRODI TEKNIK INFORMATIKA

### VISI

Menjadi Program Studi Informatika yang diakui secara internasional dan unggul dalam bidang Informatika serta berbasis nilai-nilai Islam.

### MISI

1. Menjalankan pendidikan sesuai dengan kompetensi bidang Informatika yang diakui nasional dan internasional
2. Meningkatkan penelitian dosen dan mahasiswa dalam bidang Informatika yang kreatif, inovatif dan tepat guna.
3. Meningkatkan kuantitas dan kualitas publikasi ilmiah tingkat nasional dan internasional
4. Melaksanakan dan meningkatkan kegiatan pengabdian masyarakat oleh dosen dan mahasiswa dalam bidang Informatika.
5. Menyelenggarakan aktivitas yang mendukung pengembangan program studi dengan melibatkan dosen dan mahasiswa.
6. Menyelenggarakan kerja sama dengan lembaga tingkat nasional dan internasional.
7. Menciptakan kehidupan Islami di lingkungan program studi.

## TATA TERTIB LABORATORIUM TEKNIK INFORMATIKA

### DOSEN/KOORDINATOR PRAKTIKUM

1. Dosen harus hadir saat praktikum minimal 15 menit di awal kegiatan praktikum dan menandatangani presensi kehadiran praktikum.
2. Dosen membuat modul praktikum, soal seleksi asisten, pre-test, post-test, dan responsi dengan berkoordinasi dengan asisten dan pengampu mata praktikum.
3. Dosen berkoordinasi dengan koordinator asisten praktikum untuk evaluasi praktikum setiap minggu.
4. Dosen menandatangani surat kontrak asisten praktikum dan koordinator asisten praktikum.
5. Dosen yang tidak hadir pada slot praktikum tertentu tanpa pemberitahuan selama 2 minggu berturut-turut mendapat teguran dari Kepala Laboratorium, apabila masih berlanjut 2 minggu berikutnya maka Kepala Laboratorium berhak mengganti koordinator praktikum pada slot tersebut.

### PRAKTIKAN

1. Praktikan harus hadir 15 menit sebelum kegiatan praktikum dimulai, dan dispensasi terlambat 15 menit dengan alasan yang jelas (kecuali asisten menentukan lain dan patokan jam adalah jam yang ada di Laboratorium, terlambat lebih dari 15 menit tidak boleh masuk praktikum & dianggap INHAL).
2. Praktikan yang tidak mengikuti praktikum dengan alasan apapun, wajib mengikuti INHAL, maksimal 4 kali praktikum dan jika lebih dari 4 kali maka praktikum dianggap GAGAL.
3. Praktikan harus berpakaian rapi sesuai dengan ketentuan Universitas, sebagai berikut:
  - a. Tidak boleh memakai Kaos Oblong, termasuk bila ditutupi Jaket/Jas Almamater (Laki-laki / Perempuan) dan Topi harus Dilepas.
  - b. Tidak Boleh memakai Baju ketat, Jilbab Minim dan rambut harus tertutup jilbab secara sempurna, tidak boleh kelihatan di jidat maupun di punggung (khusus Perempuan).
  - c. Tidak boleh memakai baju minim, saat duduk pun pinggang harus tertutup rapat (Laki-laki / Perempuan).
  - d. Laki-laki tidak boleh memakai gelang, anting-anting ataupun aksesoris Perempuan.
4. Praktikan tidak boleh makan dan minum selama kegiatan praktikum berlangsung, harus menjaga kebersihan, keamanan dan ketertiban selama mengikuti kegiatan praktikum atau selama berada di dalam laboratorium (tidak boleh membuang sampah sembarangan baik kertas, potongan kertas, bungkus permen baik di lantai karpet maupun di dalam ruang CPU).
5. Praktikan dilarang meninggalkan kegiatan praktikum tanpa seizin Asisten atau Laboran.
6. Praktikan harus meletakkan sepatu dan tas pada rak/loker yang telah disediakan.
7. Selama praktikum dilarang NGENET/NGE-GAME, kecuali mata praktikum yang membutuhkan atau menggunakan fasilitas Internet.
8. Praktikan dilarang melepas kabel jaringan atau kabel power praktikum tanpa sepengetahuan laboran
9. Praktikan harus memiliki FILE Petunjuk praktikum dan digunakan pada saat praktikum dan harus siap sebelum praktikum berlangsung.
10. Praktikan dilarang melakukan kecurangan seperti mencontek atau menyalin pekerjaan praktikan yang lain saat praktikum berlangsung atau post-test yang menjadi tugas praktikum.
11. Praktikan dilarang mengubah setting software/hardware komputer baik menambah atau mengurangi tanpa permintaan asisten atau laboran dan melakukan sesuatu yang dapat merugikan laboratorium atau praktikum lain.



12. Asisten, Koordinator Praktikum, Kepala laboratorium dan Laboran mempunyai hak untuk menegur, memperingatkan bahkan meminta praktikan keluar ruang praktikum apabila dirasa anda mengganggu praktikan lain atau tidak melaksanakan kegiatan praktikum sebagaimana mestinya dan atau tidak mematuhi aturan lab yang berlaku.
13. Pelanggaran terhadap salah satu atau lebih dari aturan diatas maka Nilai praktikum pada pertemuan tersebut dianggap 0 (NOL) dengan status INHAL.

### ASISTEN PRAKTIKUM

1. Asisten harus hadir 15 Menit sebelum praktikum dimulai (konfirmasi ke koordinator bila mengalami keterlambatan atau berhalangan hadir).
2. Asisten yang tidak bisa hadir WAJIB mencari pengganti, dan melaporkan kepada Koordinator Asisten.
3. Asisten harus berpakaian rapi sesuai dengan ketentuan Universitas, sebagai berikut:
  - a. Tidak boleh memakai Kaos Oblong, termasuk bila ditutupi Jaket/Jas Almamater (Laki-laki / Perempuan) dan Topi harus Dilepas.
  - b. Tidak Boleh memakai Baju ketat, Jilbab Minim dan rambut harus tertutup jilbab secara sempurna, tidak boleh kelihatan di jidat maupun di punggung (khusus Perempuan).
  - c. Tidak boleh memakai baju minim, saat duduk pun pinggang harus tertutup rapat (Laki-laki / Perempuan).
  - d. Laki-laki tidak boleh memakai gelang, anting-anting ataupun aksesoris Perempuan.
4. Asisten harus menjaga kebersihan, keamanan dan ketertiban selama mengikuti kegiatan praktikum atau selama berada di laboratorium, menegur atau mengingatkan jika ada praktikan yang tidak dapat menjaga kebersihan, ketertiban atau kesopanan.
5. Asisten harus dapat merapikan dan mengamankan presensi praktikum, Kartu Nilai serta tertib dalam memasukan/Input nilai secara Online/Offline.
6. Asisten harus dapat bertindak secara profesional sebagai seorang asisten praktikum dan dapat menjadi teladan bagi praktikan.
7. Asisten harus dapat memberikan penjelasan/pemahaman yang dibutuhkan oleh praktikan berkenaan dengan materi praktikum yang diasistensi sehingga praktikan dapat melaksanakan dan mengerjakan tugas praktikum dengan baik dan jelas.
8. Asisten tidak diperkenankan mengobrol sendiri apalagi sampai membuat gaduh.
9. Asisten dimohon mengkoordinasikan untuk meminta praktikan agar mematikan komputer untuk jadwal terakhir dan sudah dilakukan penilaian terhadap hasil kerja praktikan.
10. Asisten wajib untuk mematikan LCD Projector dan komputer asisten/praktikan apabila tidak digunakan.
11. Asisten tidak diperkenankan menggunakan akses internet selain untuk kegiatan praktikum, seperti Youtube/Game/Medsos/Streaming Film di komputer praktikan.

### LAIN-LAIN

1. Pada Saat Responsi Harus menggunakan Baju Kemeja untuk Laki-laki dan Perempuan untuk Praktikan dan Asisten.
2. Ketidakhadiran praktikum dengan alasan apapun dianggap INHAL.
3. Izin praktikum mengikuti aturan izin SIMERU/KULIAH.
4. Yang tidak berkepentingan dengan praktikum dilarang mengganggu praktikan atau membuat keributan/kegaduhan.
5. Penggunaan lab diluar jam praktikum maksimal sampai pukul 21.00 dengan menunjukkan surat ijin dari Kepala Laboratorium Prodi Teknik Informatika.

Yogyakarta, 1 Agustus 2021

Kepala Laboratorium Teknik Informatika



Lisna Zahrotun, S.T., M.Cs.

NIY. 60150773

## DAFTAR ISI

HAK CIPTA .....	1
KATA PENGANTAR.....	2
DAFTAR PENYUSUN.....	3
HALAMAN REVISI.....	4
HALAMAN PERNYATAAN.....	5
VISI DAN MISI PRODI TEKNIK INFORMATIKA .....	6
TATA TERTIB LABORATORIUM TEKNIK INFORMATIKA.....	7
DAFTAR ISI .....	10
DAFTAR GAMBAR .....	11
DAFTAR TABEL.....	12
SKENARIO PRAKTIKUM SECARA DARING .....	13
PRAKTIKUM 1: PENGENALAN <i>DEVELOPMENT ENVIRONMENT</i> .....	15
PRAKTIKUM 2: <i>INPUT DAN OUTPUT PROGRAM</i> .....	22
PRAKTIKUM 3: TIPE DATA DAN VARIABEL .....	28
PRAKTIKUM 4: OPERATOR.....	34
PRAKTIKUM 5: PERCABANGAN / KONDISIONAL.....	42
PRAKTIKUM 6: PERULANGAN .....	50
PRAKTIKUM 7: PERULANGAN BERTINGKAT.....	58
PRAKTIKUM 8: FUNGSI .....	64
PRAKTIKUM 9: FUNGSI REKURSIF .....	73
PRAKTIKUM 10: ARRAY .....	79
PRAKTIKUM 11: ARRAY 2 DIMENSI.....	86
PRAKTIKUM 12: STRUCT .....	93
PRAKTIKUM 13: POINTER .....	101
PRAKTIKUM 14: AKSES FILE .....	108
DAFTAR PUSTAKA.....	115

## DAFTAR GAMBAR

Gambar 12.1 Ilustrasi Pointer .....	102
-------------------------------------	-----

## DAFTAR TABEL

## SKENARIO PRAKTIKUM SECARA DARING

Nama Mata Praktikum : Dasar Pemrograman

Jumlah Pertemuan : 14

**TABEL SKENARIO PRAKTIKUM DARING**

Pertemuan ke	Judul Materi	Waktu (Lama praktikum sampai pengumpulan posttest)	Skenario Praktikum dari pemberian pre-test, post-test dan pengumpulannya serta mencantumkan metode yang digunakan misal video, whatsapp group, Google meet atau lainnya
1	Pengenalan Lingkungan Pengembangan	2 Hari	<ol style="list-style-type: none"> <li>1. kelas google class room dibuat sesuai jadwal praktikum yang sudah tersedia (persesi).</li> <li>2. Pret test diberikan selama 20 menit dengan menggunakan google form dan toleransi waktu 15 menit</li> <li>3. Materi praktikum didistribusikan melalui Google Class Room dalam bentuk Video atau PPT.</li> <li>4. Penyampaian materi akan dibuat video penyampaian materi dengan durasi 10 menit oleh asisten penanggung jawab.</li> <li>5. Pelaksanaan Praktikum diberikan dengan mengerjakan sesuai langkah-langkah praktikum dengan durasi 105 menit.</li> <li>6. Post Test diberikan dan diselaikan pengerjaannya dengan durasi jam melalui google class room dengan jangka waktu 2 Hari.</li> <li>7. Tugas dosen pengampu: membuat soal pre test, membuat soal response, berkoordinasi dengan setiap asisten sesuai sesi</li> <li>8. Tugas asisten: menjelaskan materi, koreksi pre test, pelaksanaan praktikum dan post test</li> </ol>
2	Pengenalan Input Output	2 Hari	SDA
3	Tipe Data dan Variabel	2 Hari	SDA
4	Operator	2 Hari	SDA
5	Percabangan/Kondisional	2 Hari	SDA
6	Perulangan	2 Hari	SDA
7	Perulangan Bertingkat	2 Hari	SDA
<b>Ujian Tengah Semester (UTS)</b>		180 Menit	Ujian Tengah Semester diberikan dan dikumpulkan melalui e-learning (Google Classroom) sesuai waktu yang ditentukan.

8	Fungsi		<ol style="list-style-type: none"> <li>1. kelas google class room dibuat sesuai jadwal praktikum yang sudah tersedia (persesi).</li> <li>2. Pret test diberikan selama 20 menit dengan menggunakan google form dan toleransi waktu 15 menit</li> <li>3. Materi praktikum didistribusikan melalui Google Class Room dalam bentuk Video atau PPT.</li> <li>4. Penyampaian materi akan dibuat video penyampaian materi dengan durasi 10 menit oleh asisten penanggung jawab.</li> <li>5. Pelaksanaan Praktikum diberikan dengan mengerjakan sesuai langkah-langkah praktikum dengan durasi 105 menit.</li> <li>6. Post Test diberikan dan diselaikan pengerjannya dengan durasi jam melalui google class room dengan jangka waktu 2 Hari.</li> <li>7. Tugas dosen pengampu: membuat soal pre test, membuat soal response, berkoordinasi dengan setiap asisten sesuai sesi</li> <li>8. Tugas asisten: menjelaskan materi, koreksi pre test, pelaksanaan praktikum dan post test</li> </ol>
9	Fungsi Rekursif		SDA
10	Library		SDA
11	Array		SDA
12	Struct		SDA
13	Pointer		SDA
14	Akses File		SDA
<b>Ujian Akhir Semester (UAS)</b>		180	Ujian Akhir Semester diberikan dan dikumpulkan melalui e-learning (Google Classroom) sesuai waktu yang ditentukan.

## PRAKTIKUM 1: PENGENALAN *DEVELOPMENT ENVIRONMENT*

Pertemuan ke : 1

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-07	Mampu memilih, membuat dan menerapkan teknik, sumber daya, penggunaan perangkat teknik modern dan implementasi teknologi informasi untuk memecahkan masalah.
CPMK-01	Mampu melakukan instalasi dan pengaturan tools pemrograman serta mengidentifikasi kebutuhan library dan komponen pemrograman

### 1.1. DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menuliskan program pada development environment yang digunakan.
2. Mengubah dan memahami statement
3. Mengcompile program
4. Menjalankan program

### 1.2. INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-07	CPMK-01	Mampu menuliskan program pada development environment yang digunakan
CPL-07	CPMK-01	<ol style="list-style-type: none"> <li>1. Mampu membuat statement sederhana pada program.</li> <li>2. Mampu melakukan kompilasi program.</li> <li>3. Mampu menjalankan program.</li> </ol>

### 1.3. TEORI PENDUKUNG

Dev C++ adalah sebuah IDE (*Integrated Development Environment*) dimana di dalamnya terdapat beberapa komponen yaitu *text editor*, *compiler*, *project builder*, *debugger*, dll. Program yang dibuat dengan bahasa C/C++ di-*compile* terlebih dahulu, sebelum dijalankan. Fungsi *compiler* adalah mengubah kode program menjadi file yang siap dieksekusi dan dalam bentuk bahasa mesin.



#### 1.4. HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

1. Komputer.
2. Dev C++

#### 1.5. PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-07	CPMK-01	Sebutkan software apa saja yang dapat digunakan untuk membuat program dengan bahasa pemrograman C/C++	30
2.	CPL-07	CPMK-01	1. Tuliskan struktur utama program pada bahasa pemrograman C/C++. 2. Bagaimana cara melakukan kompilasi program dalam bahasa C/C++ dengan software Dev C++ 3. Bagaimana cara menjalankan program pada sofware Dev C++	30 20 20

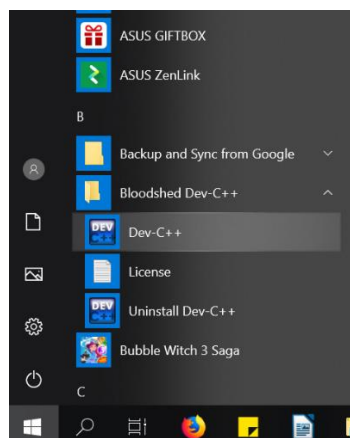
#### 1.6. LANGKAH PRAKTIKUM

Aturan Penilaian (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-07	CPMK-01	Selesaikan langkah praktikum 1 – 4	Hasil praktikum langkah 1 – 4	30
2.	CPL-07	CPMK-01	Selesaikan langkah praktikum 5-8	Hasil praktikum langkah 5-8	70

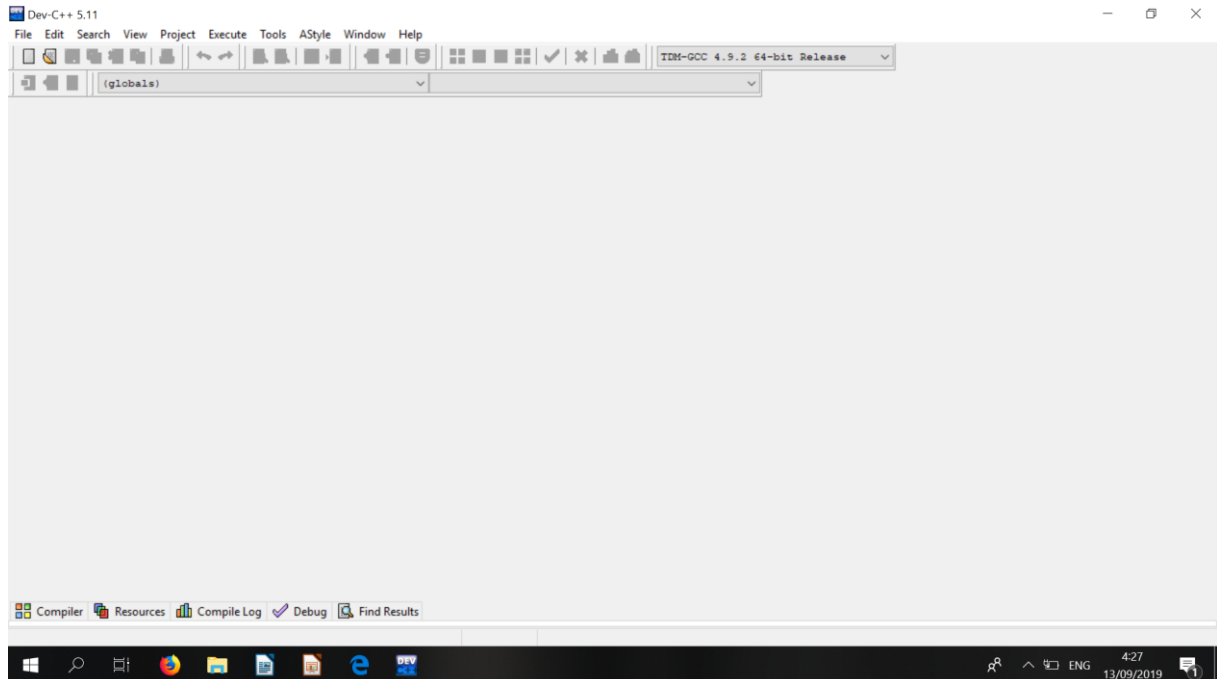
**Langkah-Langkah Praktikum:**

1. Install software Dev-C++ pada laptop atau komputer Anda. Download installer Dev=C++ dari <https://sourceforge.net/projects/orwelldevcpp/>
2. Setelah DevC++ terinstall, Anda akan mendapatkan menu Bloodshed Dev-C++ di menu utama Windows. Perhatikan Gambar 1.1 di bawah ini



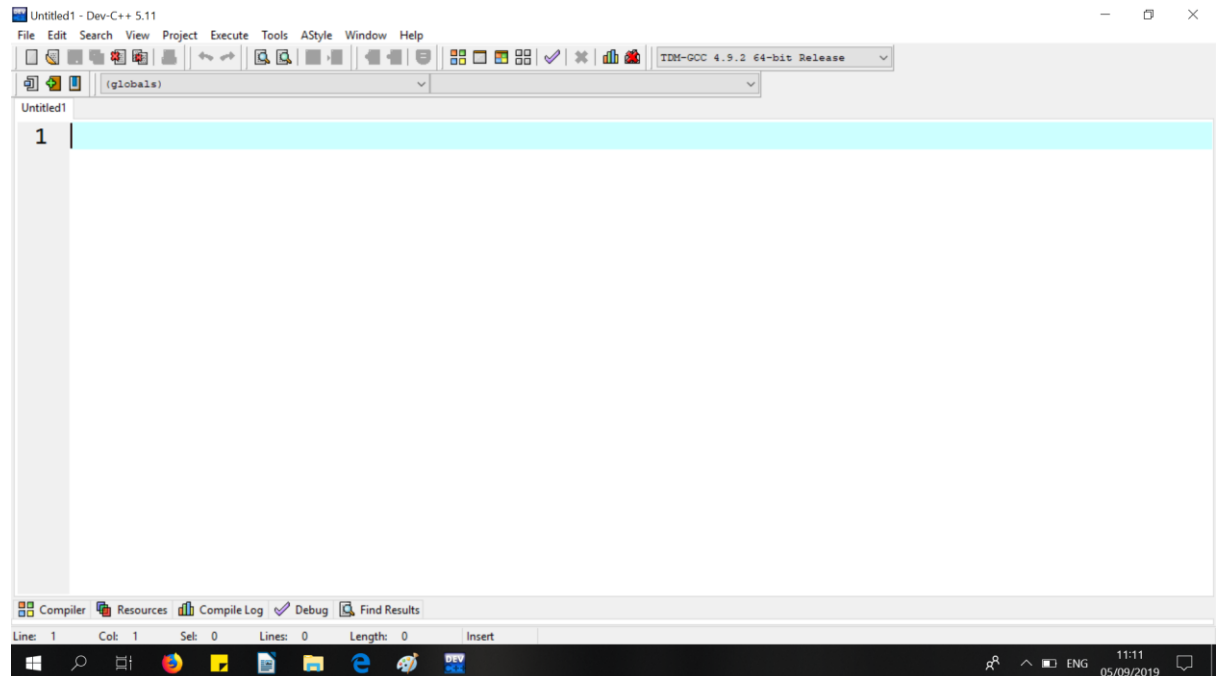
*Gambar 1.1 Tampilan DevC++ pada menu Windows*

### 3. Perhatikan tampilan aplikasi Dev C++ setelah dijalankan



Gambar 1.2. tampilan awal Dev-C++

4. Dev C++ merupakan IDE (*Integrated Development Environment* – Lingkungan Pengembangan Terintegrasi). Di dalam Dev C++ ini terdapat *text editor*, *compiler*, serta *debugger*. Berikut ini adalah tampilan dari text editor di dalam Dev C++. (klik File → New)



Gambar 1.3. Tampilan Text Editor Dev-C++

5. Sekarang kita akan mulai menuliskan kode program. Tuliskan program berikut ini di dalam text editor Dev-C++ dan simpan dalam file **Hello.cpp**. **Perhatikan setiap karakter/huruf pada kode program. Jangan sampai ada yang terlewat.**

```
/* Nama : isikan nama di sini */
```

```

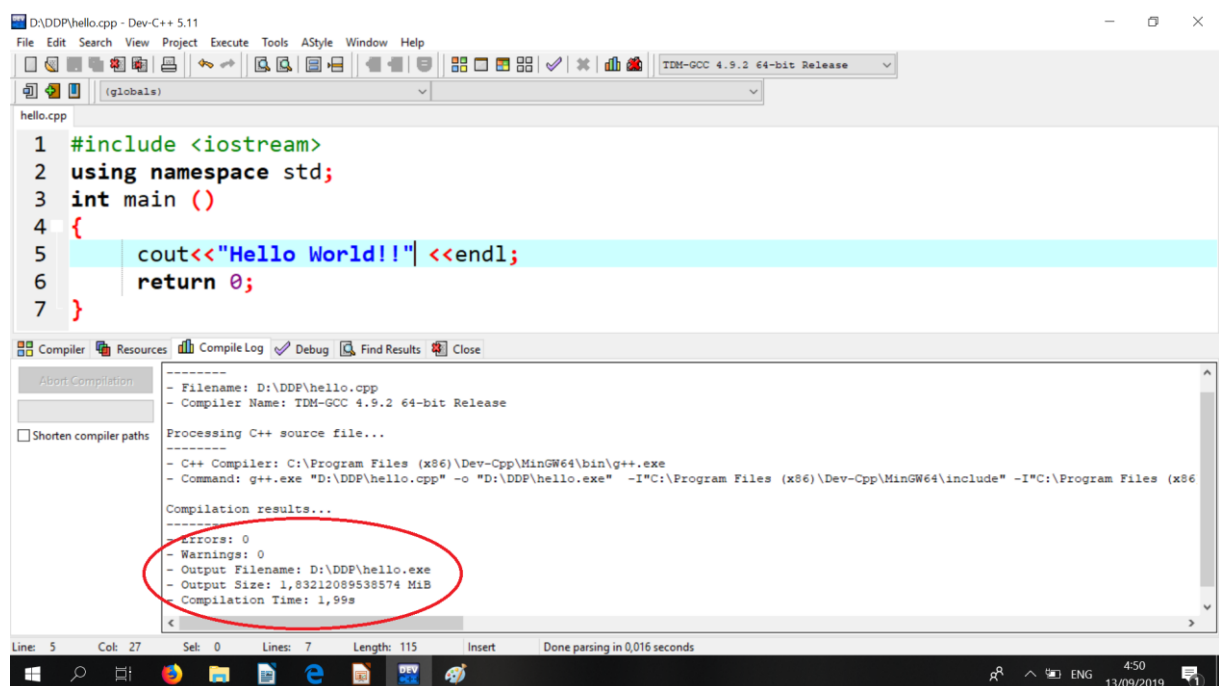
/* NIM : isikan nim di sini */

#include <iostream>
using namespace std;
int main ()
{
    cout<<"Hello World!!" <<endl;
    return 0;
}

```

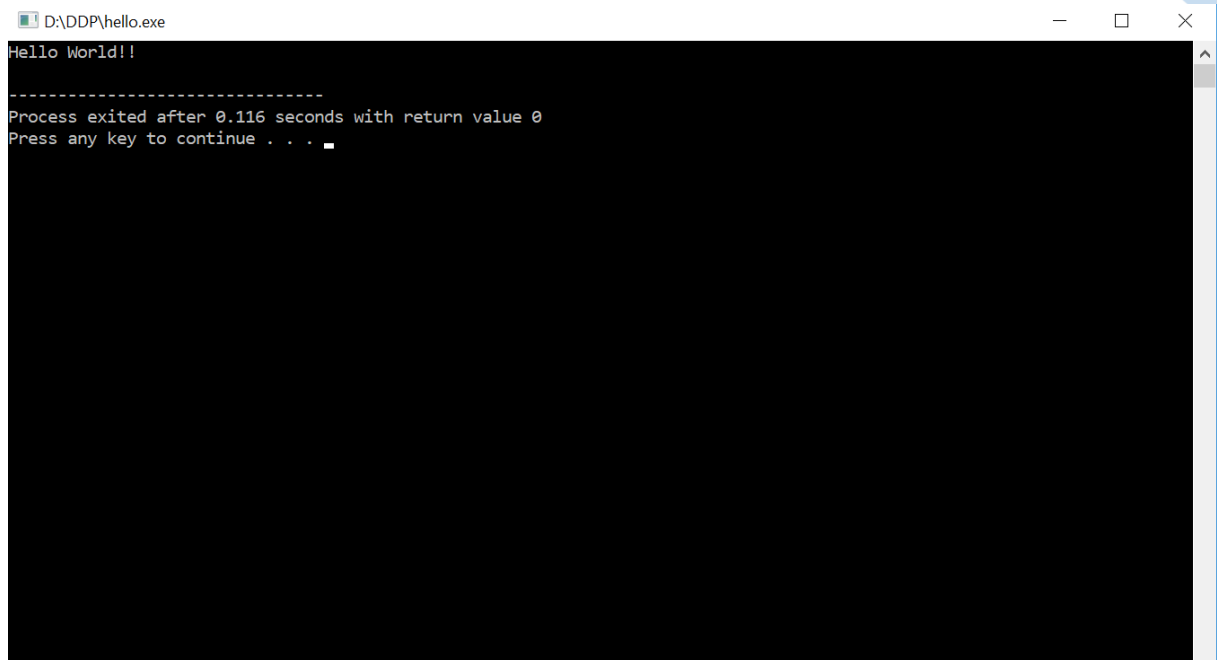
*Listing 1.1 Kode untuk menampilkan Hello World dengan bahasa C++*

6. Setelah kode Hello.cpp ditulis, lakukan Kompilasi pada kode tersebut. Klik **Execute**→ **Compile**  
 Jika kode Anda sudah benar, maka seharusnya tidak akan muncul error saat kompilasi.



*Gambar 1.4 Tampilan hasil kompilasi program*

7. Setelah Anda berhasil mengcompile program Anda, maka Anda dapat menjalankan program Anda dengan klik **Execute**→ **Run** dan lihatlah hasilnya seperti pada Gambar 1.5 di bawah ini.



Gambar 1.5 Tampilan program Hello World saat dijalankan

8. Berikut ini adalah beberapa *shortcut* penting ketika menggunakan Dev-C++

Shortcut	Fungsi
F9	Compile
F10	Run
F11	Compile and Run
	New File
	Compile
	Compile and Run
	Stop Execution (menghentikan program yang running)

## 1.7. POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-07	CPMK-01	Jelaskan file apa yang dihasilkan dari proses kompilasi kode program bahasa C/C++	30
2.	CPL-07	CPMK-01	Buatlah sebuah program sederhana untuk menampilkan tulisan: <b>Selamat Datang Di kelas Dasar-Dasar Pemrograman</b> <b>Nama: XXXX</b> Keterangan: XXXX di ganti dengan nama mahasiswa masing-masing	70

### 1.8. HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-07	CPMK-01	20%		
2.	Praktik	CPL-07	CPMK-01	50%		
3.	Post-Test	CPL-07	CPMK-01	30%		
Total Nilai						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 2: *INPUT DAN OUTPUT PROGRAM*

**Pertemuan ke** : 2

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-07	Mampu memilih, membuat dan menerapkan teknik, sumber daya, penggunaan perangkat teknik modern dan implementasi teknologi informasi untuk memecahkan masalah
CPMK-01	Mampu melakukan instalasi dan pengaturan tools pemrograman serta mengidentifikasi kebutuhan library dan komponen pemrograman

### 2.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan cara membuat program untuk input dan output sederhana
2. Menerapkan input dan output pada program

### 2.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-07	CPMK-01	<ul style="list-style-type: none"> <li>- Kemampuan mahasiswa dalam menjelaskan input, output</li> <li>- Kemampuan mahasiswa mengimplementasikan input, output pada program komputer dengan C++.</li> </ul>
--------	---------	--

### 2.3 TEORI PENDUKUNG

#### a) Library Input/Output

Pada pemrograman C++, terdapat beberapa header yang penting, antara lain:

- a. `<iostream>`: File ini mendefinisikan `cin`, `cout`, `cerr` dan `clog` yang menjadi standar input/output stream.
- b. `<iomanip>`: File ini mendeklarasikan layanan yang digunakan untuk formatting I/O, seperti `setw` dan `setprecision`.

c. `<fstream>`: File ini mendeklarasikan layanan untuk pemrosesan file.

b) Standar output stream

Cout adalah *instance* dari ostream class. Perintah cout digunakan untuk menghubungkan dengan standar perangkat device, seperti layar, printer dan sebagainya. Cout digunakan juga sebagai operator untuk penyisipan. Perintahnya diikuti dengan tanda `<<`.

c) Standar input stream

Perintah untuk proses input adalah cin sebagai *instance* dari class istream. Perintah ini digunakan untuk menerima perintah dari input device, seperti keyboard. Simbol untuk penggunaan cin adalah `>>`

## 2.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

3. Komputer.
4. Dev C++.

## 2.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-07	CPMK-01	Library atau File Header apakah yang di gunakan untuk menuliskan perintah input dan output pada program C++	40
2.	CPL-07	CPMK-01	Apa perintah yang di gunakan untuk menampilkan data dan menerima data dalam bahasa pemograman C++ dan bagaimana format penulisanya	60

## 2.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-07	CPMK-01	Selesaikan langkah praktikum 1 – 3	Hasil praktikum langkah 1 – 3	30
2.	CPL-07	CPMK-01	Selesaikan langkah praktikum 4 – 8	Hasil praktikum langkah 4 – 8	70

### Langkah-Langkah Praktikum:

1. Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
2. Ketikkan program C di bawah ini pada *text editor* IDE Dev C++, simpan dengan nama prakt2-1.c

```
#include <stdio.h>

int main() {
    /* my first program in C */
    printf("Hello, World! \n");
}
```



```
    return 0;
}
```

Perhatikan, program tersebut akan menampilkan tulisan “Hello World” pada console. Fungsi `printf` pada bahasa C berfungsi untuk meng-outputkan nilai di dalam () pada console.

3. Ketikkan program C di bawah ini pada *text editor* IDE Dev C++, simpan dalam nama `prakt2-2.cpp`

```
#include <iostream>
using namespace std;
int main() {
    cout << "Hello World" << endl;
    return 0;
}
```

Perhatikan, output program tersebut akan menampilkan tulisan “Hello World” juga pada console namun ditulis dengan bahasa C++. `cout <<` pada bahasa C++ berfungsi untuk meng-outputkan nilai di belakang `<<` pada console.

4. Ketikkan program berikut ini dan simpan dalam file `prakt2-3.cpp`

```
#include <iostream>
using namespace std;
int main() {
    int x;
    /* my program in C++ */
    cout << "Masukkan sebuah angka antara 0-100 :" << endl;
    cin >> x;
    cout << "Angka yang dimasukkan :" << x << endl;
    return 0;
}
```

Perhatikan, program di atas akan meminta user untuk memasukkan nilai, dan kemudian menampilkan nilai tersebut. `cin >>` digunakan untuk mengambil input dari pengguna/user.

5. Ketikkan program berikut ini dan simpan dalam file `prakt2-4.c`

```
#include <stdio.h>
int main() {
    int x;
    /* my program in C */
    printf("Masukkan sebuah angka antara 0-100 : \n");
```

```
scanf("%d", &x) ;
printf ("Angka yang dimasukkan = %d",x);
return 0;
}
```

Perhatikan, program di atas akan meminta user untuk memasukkan nilai, dan kemudian menampilkan nilai tersebut. Fungsi `scanf` digunakan untuk mengambil input dari pengguna/user.

6. Ketikkan program C++ berikut pada IDE yang disediakan, simpan dengan nama `prakt2-5.cpp`. Jalankan program `prakt2-5.cpp` Tuliskan apakah yang muncul saat program ini dijalankan?

```
#include <iostream>
using namespace std;
int main() {
    char str[] = "Hello C++";
    cout << "Nilai string ini adalah : " << str << endl;
    return 0;
}
```

7. Ketikkan program C++ berikut, simpan dengan nama `prakt2-6.cpp` kemudian jalankan.

```
#include <iostream>
using namespace std;
int main() {
    char name[50];
    cout << "Silahkan masukkan nama anda: ";
    cin >> name;
    cout << "Namamu adalah: " << name << endl;
    return 0;
}
```

`cout<<` digunakan untuk menampilkan nilai di belakangnya pada console  
`cin >>` digunakan untuk mengambil nilai input yang diberikan oleh pengguna/user

8. Buatlah program `prakt.2.7.cpp` yang menggunakan input dan output untuk memasukkan dan menampilkan biodata lengkap anda menggunakan Bahasa C++.

## 2.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-07	CPMK-01	Buatlah program untuk memasukkan biodata sesuai dengan data KTP/ SIM, kemudian buatlah tampilannya semirip mungkin dengan tampilan KTP/ SIM anda (tanpa foto).	70
2	CPL-07	CPMK-01	Buat analisa dan penjelasan maksud dari program diatas	30

## 2.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-07	CPMK-01	20%		
2.	Praktik	CPL-07	CPMK-01	50%		
3.	Post-Test	CPL-07	CPMK-01	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 3: TIPE DATA DAN VARIABEL

**Pertemuan ke** : 3

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-02	Mampu mengimplementasi program terstruktur sesuai dengan spesifikasi program menggunakan perintah eksekusi bahasa pemrograman

### 3.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan jenis-jenis tipe data dan variabel serta penggunaannya
2. Menerapkan pembuatan variabel dengan jenis tipe data yang sesuai kedalam sebuah program komputer dengan C++.

### 3.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-02	Kemampuan mahasiswa dalam menjelaskan jenis-jenis tipe data dan memilih tipe data yang tepat pada sebuah variabel
CPL-03	CPMK-02	Kemampuan mahasiswa dalam menerapkan pembuatan variabel dengan tipe data yang sesuai pada program komputer dengan C++.

### 3.3 TEORI PENDUKUNG

Variabel

Variabel merupakan suatu nama/pengenal yang digunakan untuk menampung suatu nilai dalam program. Penentuan nama variable tersebut dalam bahasa pemrograman C/C++ harus mengikuti aturan antara lain:

- a. Pengenal harus diawali dengan huruf (A..Z, a..z) atau karakter garis bawah ( \_ ). Selanjutnya dapat berupa huruf, digit (0..9) atau karakter garis bawah atau tanda dollar (\$).
- b. Panjang pengenal boleh lebih dari 31 karakter, tetapi hanya 31 karakter pertama yang akan dianggap berarti.
- c. Pengenal tidak boleh menggunakan nama yang tergolong sebagai kata- kata cadangan (reserved words) seperti main, include, int, float, char, dan sebagainya.
- d. Variabel yang akan digunakan dalam program harus dideklarasikan terlebih dahulu sesuai dengan tipe data yang akan digunakan.

#### Jenis/Tipe Data

Penentuan tipe data merupakan hal yang sangat penting dalam suatu pemrograman. Tipe data yang berbeda dengan operasi yang sama bisa menghasilkan data yang berbeda. Misal data 7 dibagi 4 dengan tipe integer akan menghasilkan 1, namun jika tipe datanya float maka akan menghasilkan nilai 1.75000000. Sehingga pemilihan tipe data yang tepat sangat diperlukan karena akan menentukan efektifitas/efisiensi dalam pemrograman maupun hasilnya. Tipe data dalam bahasa pemrograman C/C++ adalah int, float, double, long, char, string.

### 3.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

- 4 Komputer.
- 5 Dev C++.

### 3.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	Jelaskan apa yang dimaksud dengan tipe data int, float, double, dan char.	40
2.	CPL-03	CPMK-02	Bagaimana sintak untuk mendeklarasikan sebuah variable yang digunakan menyimpan data bilangan real, memberikan nilai pada variabel dengan input data user, dan mencetak nilai variabel tersebut.	60

### 3.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-02	Selesaikan langkah praktikum 1 – 4	Hasil praktikum langkah 1 – 4	60
2.	CPL-03	CPMK-02	Selesaikan langkah praktikum 5	Hasil praktikum langkah 5	40

### Langkah-Langkah Praktikum:

1. Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
2. Ketikkan program berikut menggunakan Bahasa C. Jalankan kemudian lakukan analisa apa perbedaan hasil ouput kode baris 7 dengan 8, 9 dengan 10, 11 dengan 12, 13 dengan 14 dengan 15 dengan 16.

```
/* File program : prak.3.1.c */

#include <stdio.h>

int main()

{

    int Be = 345,da=-768;

    float la= 123.4567890123;

    printf("%d \n",Be);

    printf("%i \n",Be);

    printf("10%d \n",da);

    printf("10%u \n",da);

    printf("%hd \n",Be);

    printf("%ld \n",Be);

    printf("%f \n",la);

    printf("%10.5f \n",la);

    printf("%e \n",la);

    printf("%g \n",la);

    printf("Perkalian Be x la = %10.3f \n",Be*la);

    printf("Pembagian da / Be = %10.5f \n",da/Be);

    return 0;

}
```

3. Ketikkan program berikut dengan Bahasa C kemudian jalankan dan lakukan analisis. Apa yang terjadi jika baris ke 8 diganti dengan 'c = 0.5f\*d+a;' ? Jelaskan pendapat anda!

```
/* File program : prak3-2.c */
```

```

#include <stdio.h>

int main()
{
    float a,b ;
    int d;
    char c;
    c='B'; d=5 ; a= 9.25;

    b=0.5f*d+a;

    printf("Nilai %c = %b ",c,b);

    return 0;
}

```

4. Ketikkan program berikut dengan Bahasa C kemudian jalankan dan lakukan analisis, apa yang dihasilkan dari penjumlahan terhadap variabel float, double, dan integer.

```

/* File program: prak3-3.c */

#include<stdio.h>
#include<string.h>
int main() {
    float x = 10.327;
    double y = 4244.546;
    int z = 28;
    printf("The float value : %f\n", x);
    printf("The double value : %f\n", y);
    printf("The sum of float,
    double and int variable : %f\n", (x+y+z));
    return 0;
}

```

5. Ketikkan program berikut ini dengan bahasa C kemudian jalankan. Perhatikan bahwa kode program tersebut mengecek nilai minimum dan nilai maksimum pada berbagai tipe data, juga ukuran untuk setiap tipe data.

```

/* File Program : prak3-4.c */

#include <stdio.h>
#include <stdlib.h>
#include <limits.h>
#include <float.h>

```



```

int main() {

    printf("CHAR_BIT      :   %d\n", CHAR_BIT);
    printf("CHAR_MAX      :   %d\n", CHAR_MAX);
    printf("CHAR_MIN      :   %d\n", CHAR_MIN);
    printf("Storage size for int : %d \n", sizeof(int));
    printf("INT_MAX       :   %d\n", INT_MAX);
    printf("INT_MIN       :   %d\n", INT_MIN);
    printf("Storage size for long : %d \n", sizeof(long));
    printf("LONG_MAX      :   %ld\n", (long) LONG_MAX);
    printf("LONG_MIN      :   %ld\n", (long) LONG_MIN);

    printf("Storage size for float : %d \n", sizeof(float));
    printf("FLT_MAX       :   %g\n", (float) FLT_MAX);
    printf("FLT_MIN       :   %g\n", (float) FLT_MIN);
    printf("-FLT_MAX      :   %g\n", (float) -FLT_MAX);
    printf("-FLT_MIN      :   %g\n", (float) -FLT_MIN);
    printf("DBL_MAX       :   %g\n", (double) DBL_MAX);
    printf("DBL_MIN       :   %g\n", (double) DBL_MIN);
    printf("-DBL_MAX      :   %g\n", (double) -DBL_MAX);
    printf("Precision value: %d\n", FLT_DIG );

    return 0;
}

```

### 3.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	Pada program post test praktikum sebelumnya (Program KTP/SIM), Lakukan analisa berapa variabel yang di perlukan dan apa tipe data variabel tersebut, tuliskan alasanya kenapa memilih tipe data tersebut.	50
2.	CPL-03	CPMK-02	Modifikasi program KTP/SIM dari praktikum sebelumnya dengan masing-masing variabel yang telah disesuaikan	50

### 3.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-02	20%		
2.	Praktik	CPL-03	CPMK-02	50%		
3.	Post-Test	CPL-03	CPMK-02	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 4: OPERATOR

Pertemuan ke : 4

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-02	Mampu mengimplementasi program terstruktur sesuai dengan spesifikasi program menggunakan perintah eksekusi bahasa pemrograman

### 4.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan jenis-jenis operator dan fungsinya
2. Menerapkan berbagai jenis operator pada program sesuai dengan kebutuhannya.

### 4.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-02	Kemampuan mahasiswa dalam menjelaskan dan menerapkan berbagai jenis operator (assignment, increament, decreament, aritmatika, logika, relasional) pada program C++.
--------	---------	---

### 4.3 TEORI PENDUKUNG

Operator merupakan symbol yang digunakan dalam bahasa pemrograman untuk melakukan operasi atau manipulasi terhadap data atau variable. Dalam Bahasa C dilihat dari jumlah operand, operator dapat bersifat binary atau unary. Operator binary merupakan operator yang memiliki dua operan. Operator unary merupakan operator yang memiliki satu operan.

Operator untuk operasi **aritmatika** yang tergolong sebagai operator binary adalah :

\* perkalian

- /      pembagian
- %      modulus (siswa pembagian )
- +      penjumlahan
- pengurangan

Adapun operator yang tergolong sebagai operator unary.

- tanda minus
- +      tanda plus
- ++    operator kenaikan (increment)
- operator penurunan (decrement)

Contoh `i=i+` dapat juga ditulis `i++` `j= j-1` dapat juga ditulis `j--`

Urutan pengerjaan (prioritas) operator **aritmatika** dalam bahasa C adalah seperti berikut :

Prioritas	Operator	Urutan Pengerjaan
Tertinggi	( )	dari kiri ke kanan
	! ++ -- + -	dari kanan ke kiri
	* / %	dari kiri ke kanan
	+ -	dari kiri ke kanan
Terendah	= += -= *= /= %=	dari kanan ke kiri

Selain operator aritmatika, ada pula operator **relasional** dan operator **logika**. Operator relasional dan operator logika juga termasuk ke dalam operator binary, karena memiliki dua operand. Operator relasional digunakan untuk membandingkan nilai-nilai dari operandnya, menghasilkan suatu nilai boolean True atau False. Berikut ini adalah operator-operator relasional:

- <      (kurang dari)
- <=    (kurang dari sama dengan)
- >      (lebih dari)

>=	(lebih dari sama dengan)
!=	(tidak sama dengan)
==	(sama dengan)

Berikut ini adalah operator logika:

Operator	makna/arti
&&	and
	or
!	not

#### 4.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

5. Komputer.
6. Dev C++

#### 4.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	Sebutkan dan jelaskan jenis-jenis operator dan berikan contohnya.	50
2.	CPL-03	CPMK-02	Berapa nilai X dari operasi berikut ini: a. $X=2+3*4+2$ b. $X=3<2$ c. $X=X++;$	50

#### 4.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-02	Selesaikan langkah praktikum langkah 1 – 4	Hasil praktikum langkah 1 – 4	25
2.	CPL-03	CPMK-02	Selesaikan langkah praktikum langkah 5 – 6	Hasil praktikum langkah 5 – 6	25
3	CPL-03	CPMK-02	Selesaikan langkah praktikum langkah 7	Hasil praktikum langkah 7	50

#### Langkah-Langkah Praktikum:

1. Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
2. Ketikkan program dengan Bahasa C berikut, jalankan dan amati outputnya. Lakukan percobaan dengan melakukan modifikasi pada penggunaan operator, misalnya menambahkan tanda minus (-) pada salah satu operand.

```

/* File program : prak4-1.cpp */
#include <stdio.h>
main()
{
    int    x=5;
    float  y;
    y=0.5f*x*x+2*x-5;
    printf("Untuk nilai x = %d \n",x);
    printf("Nilai Y = %f\\",y);
    printf ("Nilai x modulus 3 = %d", x%3);
}

```

3. Ketikkan program C++ berikut, jalankan program dan catat hasilnya. Lakukan perubahan nilai data inputan.

```

/* File program : prak.4-2.cpp */
#include<iostream>
using namespace std;
int main()
{
    int a,b,c; int x,y,z;
    cout<<" Nilai a : ";cin>>a;
    cout<<" Nilai b : ";cin>>b;
    cout<<" Nilai c : ";cin>>c;
    x= 2+a*3+b+c-2*2;
    y= (2+a)*3+b+c-2*2;
    z= 2+a*3+b+(c-2)*2;
    cout<<"Nilai a : "<<a;
    cout<<"\nNilai b : "<<b;
    cout<<"\nNilai c : "<<c;
    cout<<"\nHasil x : "<<x;
    cout<<"\nHasil y : "<<y;
    cout<<"\nHasil z : "<<y;
    getch();
}

```

4. Ketikkan program C++ berikut, jalankan program dan catat hasilnya. Lakukan perubahan nilai data inputan.

```
/* File program : prak.4-3.cpp */
#include<iostream>
using namespace std;
int main()
{
    float a,b,c;
    cout<<" Nilai a : ";cin>>a;
    a++;
    cout<<" Nilai b : ";cin>>b;
    b+=2; c=2*b;
    c/=a;
    cout<<"Nilai a : "<<a;
    cout<<"\nNilai b : "<<b;
    cout<<"\nNilai c : "<<c;
    getch();
}
```

5. Ketikkan program C++ berikut, jalankan program dan perhatikan hasilnya.

```
/* File program : prak.4-4.cpp */
#include<iostream>
#include<conio>
main()
{   int a, b, c;
    bool s;
    a= 15;
    b= 3*5;
    c = 10;
    s= ((a==b) || (b<c))
    cout<<(a<b)<<endl;
    cout<<(a<=b)<<endl;
```

```

        cout<<(b>c)<<endl;
        cout<<(b>=c)<<endl;
        cout<<(a!=b)<<endl;
        cout<<(a==b)<<endl;
        cout<<(b==c)<<endl;

        cout<< s <<endl;

    }

```

6. Ketikkan program C++ berikut, jalankan program dan perhatikan hasilnya.

```

/* File program : prak.4-5.cpp */

#include<iostream>

#include<conio>

#define phi 3.14159

main()

{   double r=10;                // jari-jari

    double keliling;

    keliling = 2 * phi * r;
    cout << "Keliling=" <<keliling;

}

```

7. Buatlah sebuah program untuk melakukan konversi suhu dari celcius ke Reamur dan Fahrenheit. Perhatikan penggunaan tipe data pada inputan dan hasil konversi. Buatlah output tertata rapi.

#### 4.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	Buatlah program untuk melakukan operasi penambahan, pengurangan, perkalian, dan pembagian dua buah bilangan. Dua buah bilangan merupakan nilai yang di inputkan oleh user.	40
2.	CPL-03	CPMK-02	Buatlah program untuk mengkonversi suatu nilai waktu menjadi jam, menit, dan detik. Input nilai waktu dalam sastuan detik. Contoh: <b>Masukan Jumlah waktu (Detik):5000</b> <b>Hasil Konversi: 2 jam: 23 menit:20 detik</b>	60



#### 4.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-02	20%		
2.	Praktik	CPL-03	CPMK-02	50%		
3.	Post-Test	CPL-03	CPMK-02	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 5: PERCABANGAN / KONDISIONAL

**Pertemuan ke** : 5

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-02	Mampu mengimplementasi program terstruktur sesuai dengan spesifikasi program menggunakan perintah eksekusi bahasa pemrograman

### 5.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep pernyataan kondisional dengan if, if –else, switch-case, dan kondisional bertingkat.
2. Menerapkan pernyataan kondisional (if, if-else, switch-case, kondisional bertingkat) pada bahasa pemrograman C++.

### 5.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-02	Kemampuan mahasiswa dalam menjelaskan dan menerapkan pernyataan kondisional if, if-else, switch-case, dan pernyataan kondisional bertingkat.
--------	---------	--

### 5.3 TEORI PENDUKUNG

Dalam implementasi pemrograman banyak masalah yang harus diselesaikan dengan cara memilih/mengambil keputusan. Statemen yang disediakan dalam Bahasa C / C++ ada 3 statement yaitu :

- a. Statement IF : untuk melakukan pemilihan / pemanggilan keputusan, memiliki statement dasar seperti berikut:

```
if (test-condition)
statement
```

- b. Statement IF ... ELSE : untuk melakukan pemilihan minimal 2 alternatif, memiliki statement dasar seperti berikut:

```
if (test-condition)
statement1
else
statement2
```

- c. Statement Switch : digunakan untuk melakukan pemilihan diantara beberapa pilihan yang disediakan, memiliki statement dasar seperti berikut

```
switch (integer-expression)
{
case label1 : statement(s)
case label2 : statement(s)
...
default : statement(s)
}
```

- d. *Logical Operator* : sebagai operator dalam pemilihan /pengontrolan

Tabel 5.1 operator logika dalam bahasa C/C++

Operator	makna/arti
&&	and
	or
!	not

## 5.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

- 6 Komputer.
- 7 Dev C++.

## 5.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	Jelaskan pengertian pernyataan kondisional atau percabangan	30
2.	CPL-03	CPMK-02	Jika terdapat pernyataan dengan 4 kondisi maka, tuliskan sintak percabangan dengan pernyataan if else dan switch case.	70

## 5.6 LANGKAH PRAKTIKUM

Aturan Penilaian (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-02	Selesaikan langkah praktikum 1 – 10	Hasil praktikum langkah 1 – 10	35

2.	CPL-03	CPMK-02	Selesaikan langkah praktikum 11 – 16	Hasil praktikum langkah 11 – 16	35
3.	CPL-03	CPMK-02	Selesaikan langkah praktikum 17	Hasil praktikum langkah 17	30

### Langkah-Langkah Praktikum:

- Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
- Tuliskan program berikut sebagai file baru (prak5-1.cpp)

```
#include <iostream>
int main()
{
    const float LIMIT = 60.0;
    int idMobil;
    float kecepatan;

    cout << "\nMasukkan nomor mobil dan kecepatan mobil: ";
    cin >> idMobil >> kecepatan;

    if(kecepatan > LIMIT)
        cout << "Mobil " << idNum << " melebihi batas kecepatan.\n";

    cout << "Akhir dari output program.\n";
    return 0;
}
```

*Listing 5.1 program menggunakan if*

- Jalankan program tersebut secara berulang dengan beberapa nilai input misalnya 10, 25, 50, 75. Perhatikan korelasi input dan outputnya
  - Rubah batas kecepatan menjadi suatu bilangan. Jalankan kembali program tersebut dan masukkan nilai yang telah dicoba pada langkah sebelum ini. Perhatikan korelasi input dan outputnya
- Tuliskan program berikut sebagai file baru (prak5-2.cpp)

```
#include<iostream>
int main()
{
    int x;
    cout<<"Nilai X = ";
    cin>>x;
    if (x % 2 == 0)
        cout<<"Genap";
    else
        cout<<"Ganjil";

    return 0;
}
```

*Listing 5.2 program menggunakan if-else*

- Jalankan program tersebut secara berulang dengan beberapa nilai input yang berbeda. Perhatikan korelasi input dan outputnya  
 Tambahkan baris program untuk menampilkan nilai  $x \% 2$ . Contoh  
 Input : Nilai X = 5  
 Output : Ganjil

Nilai modulus adalah 1

5. Tuliskan program berikut sebagai file baru (prak5-3.cpp)

```
#include<iostream>
int main()
{
    int angka,pembagi;
    cout << "Masukkan angka: ";
    cin >> angka >> pembagi;
    if (n%d == 0)
        cout << angka << " dapat dibagi dengan " << pembagi << endl;
    else
        cout << angka << " tidak dapat dibagi dengan " << pembagi <<
endl;

    return 0;
}
```

*Listing 5.3 program menggunakan if*

6. Jalankan program tersebut secara berulang dengan beberapa nilai input yang berbeda. Perhatikan korelasi input dan outputnya
7. Tambahkan kode pada program di atas untuk melakukan pengecekan terhadap pembagi. Nilai pembagi tidak boleh nol karena sebuah bilangan tidak dapat dibagi nol
8. Tuliskan program berikut sebagai file baru (prak5-4.cpp)

```
#include<iostream>
int main()
{
    cout << "Pilih angka dari 1 sampai 8." << endl;
    char jawaban;
    cout << "Apakah angkanya kurang dari 5? (y|t): ";
    cin >> answer;
    if (answer == 'y') // 1 <= n <= 4
    {
        cout << "Apakah angkanya kurang dari 3? (y|t): ";
        cin >> jawaban;
        if (jawaban == 'y') // 1 <= n <= 2
        {
            cout << " Apakah angkanya kurang dari 2? (y|n): ";
            cin >> jawaban;
            if (jawaban == 'y')
                cout << "Angkamu adalah 1." << endl;
            else
                cout << " Angkamu adalah 2." << endl;
        }
        else // 3 <= n <= 4
        {
            cout << " Apakah angkanya kurang dari 4? (y|t): ";
            cin >> jawaban;
            if (jawaban == 'y')
                cout << " Angkamu adalah 3." << endl;
            else
                cout << " Angkamu adalah 4." << endl;
        }
    }

    Return 0;
}
```

*Listing 5.4 program menggunakan if-else bertingkat*

9. Jalankan program tersebut secara berulang dengan beberapa nilai input yang berbeda. Perhatikan korelasi input dan outputnya
10. Tambahkan kode pada program di atas untuk melakukan pengecekan angka 5 sampai 8!

**Contoh:**

```
Pilih angka dari 1 sampai 8.
7
Apakah angkanya kurang dari 5? (y|t):
t
Apakah angkanya kurang dari 7? (y|t):
t
Apakah angkanya kurang dari 8? (y|t):
y
Angkamu adalah 7.
```

11. Tuliskan program berikut sebagai file baru (prak5-5.cpp)

```
#include<iostream>
int main()
{
    int number;
    cout<<"Enter a positive integer number: ";
    cin>>number;

    switch (number)
    {
        case 1:
            cout<< "One!!";
            break;
        case 2:
            cout<< "Two!!";
            break;
        case 3:
            cout<< "Three!!";
            break;
        default:
            cout<< "Others";
            break;
    }
    return 0;
}
```

*Listing 5.5 program menggunakan switch-case*

12. Jalankan program tersebut secara berulang dengan beberapa nilai input yang berbeda (angka dari 1 sampai 9). Perhatikan korelasi input dan outputnya
13. Tambahkan pengecekan untuk input angka 4 hingga 9. Jalankan kembali program dan masukkan beberapa nilai input yang berbeda. Perhatikan korelasi input dan outputnya
14. Tuliskan program berikut sebagai file baru (prak5-6.cpp)

```
#include<iostream>
int main()
{
    int skor;
    cout << "Masukkan skormu (0-100): ";
    cin >> skor;
    switch (skor /10)
    {
        case 10:
        case 9: cout << "Nilaimu adalah A." << endl;
            break;
```

```

case 8: cout << "Nilaimu adalah B." << endl;
        break;
case 7: cout << " Nilaimu adalah C." << endl;
        break;
case 6: cout << " Nilaimu adalah D." << endl;
case 5:
case 4:
case 3:
case 2:
case 1:
case 0: cout << " Nilaimu adalah E." << endl;
        break;
}

```

Listing 5.6 program menggunakan logical operator

15. Jalankan program tersebut secara berulang dengan beberapa nilai input yang berbeda. Perhatikan korelasi input dan outputnya
16. Ubahlah kode di atas dengan merubah penilaian dengan batas nilai E adalah 50
17. Ubahlah kode program di atas menjadi pernyataan if else dengan ketentuan nilai  $\geq 85$  adalah A,  $70 \leq \text{nilai} < 85$  adalah B,  $55 \leq \text{nilai} < 70$  adalah C,  $40 \leq \text{nilai} < 55$  adalah D, dan nilai  $< 40$  adalah E.

## 5.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor																
1.	CPL-03	CPMK-02	<p>Buatlah program untuk melakukan konversi suhu menggunakan pengontrolan dan percabangan dengan ketentuan sebagai berikut:</p> <p>Tabel perhitungan suhu sebagai berikut:</p> <table><tr><th></th><th>Celcius</th><th>Kelvin</th><th>Fahrenheit</th></tr><tr><th>Celcius</th><td><math>C = C</math></td><td><math>K = C + 273</math></td><td><math>F = 9/5 * C + 32</math></td></tr><tr><th>Kelvin</th><td><math>C = K - 273</math></td><td><math>K = K</math></td><td><math>F = 9/5 * (C - 273) + 32</math></td></tr><tr><th>Fahrenheit</th><td><math>C = 5/9 * (F - 32)</math></td><td><math>K = 5/9 * (F - 32) + 273</math></td><td><math>F = F</math></td></tr></table> <p>Pengguna memasukkan skala dan suhu asal kemudian memilih skala tujuan</p> <p>Contoh (warna merah menunjukkan nilai yang diinputkan pengguna)</p> <p>Input :</p> <p>Masukkan skala asal = F</p> <p>Masukkan suhu asal dalam F = 86</p> <p>Masukkan skala tujuan = C</p> <p>Output:</p> <p>86 derajat F sama dengan 30 derajat C</p>		Celcius	Kelvin	Fahrenheit	Celcius	$C = C$	$K = C + 273$	$F = 9/5 * C + 32$	Kelvin	$C = K - 273$	$K = K$	$F = 9/5 * (C - 273) + 32$	Fahrenheit	$C = 5/9 * (F - 32)$	$K = 5/9 * (F - 32) + 273$	$F = F$	
	Celcius	Kelvin	Fahrenheit																	
Celcius	$C = C$	$K = C + 273$	$F = 9/5 * C + 32$																	
Kelvin	$C = K - 273$	$K = K$	$F = 9/5 * (C - 273) + 32$																	
Fahrenheit	$C = 5/9 * (F - 32)$	$K = 5/9 * (F - 32) + 273$	$F = F$																	
			Buatlah program dengan pernyataan if-else	50																
			Buatlah program dengan pernyataan switch-case	50																



## 5.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-02	20%		
2.	Praktik	CPL-03	CPMK-02	50%		
3.	Post-Test	CPL-03	CPMK-02	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 6: PERULANGAN

Pertemuan ke : 6

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-02	Mampu mengimplementasi program terstruktur sesuai dengan spesifikasi program menggunakan perintah eksekusi bahasa pemrograman

### 6.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep perulangan dengan for, while, dan do while
2. Menerapkan konsep perulangan dengan for, while, dan do while pada program komputer dengan C++

### 6.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-02	Kemampuan mahasiswa dalam menjelaskan dan menerapkan konsep perulangan dengan for, while, dan do while kedalam sebuah program C++
--------	---------	---

### 6.3 TEORI PENDUKUNG

Dalam implementasi pemrograman banyak masalah yang harus diselesaikan secara berulang. Untuk menghindari penulisan statement tersebut secara berulang maka digunakan statement perulangan / *looping*. Statement yang disediakan dalam Bahasa C / C++ untuk perulangan ini ada 3 yaitu:

a. **For**, bentuk umum statement ini adalah :

```
for (expression1; expression2; expression3)
    statement;
```

**Expression1**: inisialisasi nilai awal variabel

**Expression2:** kontrol kondisi perulangan, biasanya merupakan batas akhir perulangan, ekspresi ini biasanya dalam bentuk tes kondisi

**Expression3:** perubahan yg akan dijalankan dari nilai awal variabel sampai akhir, atau merupakan step

**b. While,** bentuk umum statement ini adalah:

```
while (condition)
statement;
```

condition : seperti halnya dalam Statement IF, condition merupakan ekspresi yg memberikan nilai kembalian benar atau salah

Selama condition benar, maka statement akan dilakukan

Setelah kondisi tidak terpenuhi maka melanjutkan statement diluar perulangan

**c. Do ..While,** secara umum bentuk statement dasar dari

```
do {
statement;
} while (condition);
```

Perulangan do...while sama halnya dengan perulangan while, hanya saja pengecekan kondisi dilakukan setelah pengerjaan pertama, sehingga dengan perulangan do...while pasti dilakukan minimal 1 kali apapun kondisinya

## 6.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

1. Komputer.
2. Dev C++

## 6.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	Jelaskan apa yang di maksud dengan perulangan dan tuliskan sintak dari pernyataan for, while, dan do while.	40
2.	CPL-03	CPMK-02	Jelaskan apa perbedaan perulangan dengan for, while, dan do while.	60

## 6.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-02	Selesaikan langkah praktikum 1 – 6	Hasil praktikum langkah 1 – 6	30
2.	CPL-03	CPMK-02	Selesaikan langkah praktikum 7 – 14	Hasil praktikum langkah 7 – 14	30
3	CPL-03	CPMK-02	Selesaikan langkah praktikum 15 – 17	Hasil praktikum langkah 1 – 6	10

4	CPL-03	CPMK-02	Selesaikan langkah praktikum 18 – 23	Hasil praktikum langkah 1 – 6	30
---	--------	---------	--------------------------------------	-------------------------------	----

### Langkah-Langkah Praktikum:

- Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
- Tuliskan program berikut sebagai file baru (prak6-1.cpp)

```
#include <iostream>
int main()
{
    int c;    float r,f;
    cout << "-----";
    cout << "\n";
    cout << "Celcius \t Reamur \t Fahrenheit \n";
    for(c = 1; c < 30; c++)
        cout << "-"; /* dikerjakan dlm perulangan c */
    cout<<"\n";
    for(c = 10; c < 100; c+= 10)
    {
        r = c * 10;
        f = c + 32;
        cout << c << " \t " << r << " \t " << f << "\n";
    }

    for(c = 1; c < 30; c++)
        cout << "-"; /* dikerjakan dlm perulangan c */

    return 0;
}
```

*Listing 6.1 program menggunakan for*

- Jalankan program tersebut dan perhatikan keluarannya
- Tuliskan program berikut sebagai file baru (prak6-2.cpp)

```
#include <iostream>
int main()
{
    int n,i;
    for ( n = 0, i = 10 ; n != i ; n++, i-- )
    {
        cout << "Nilai n = " << n << " -- ";
        cout << "Nilai i = " << i << endl;
    }

    return 0;
}
```

*Listing 6.2 program menggunakan for*

- Jalankan program tersebut, dan perhatikan korelasi input dan outputnya
- Perhatikan bahwa ekspresi 1 dan ekspresi 3 ada 2 statement yang dipisahkan dengan koma.  
Ekspresi 1: `n = 0, i = 10` berarti bahwa deklarasi nilai awal `n = 0` dan `i = 10`  
Ekspresi 3: `n++, i--` berarti bahwa nilai `n` incremen 1 dan `i` decremen 1
- Tuliskan program berikut sebagai file baru (prak6-3.cpp)

```
#include <iostream>
#include <math.h>
int main()
```

```

{
    double x;
    cout << "Masukkan bilangan positif = ";
    cin >> x;
    while (x > 0)
    {
        cout << "sqrt(" << x << ") = " << sqrt(x) << endl;
        cout << "Masukkan bilangan positif lainnya (atau masukkan 0
untuk keluar): ";
        cin >> x;
    }
    return 0;
}

```

*Listing 6.3 program menggunakan while*

8. Jalankan program tersebut secara berulang dengan nilai  $x$  diubah-ubah, dan perhatikan korelasi input dan outputnya
9. Modifikasi kode untuk mengakhiri perulangan apabila pengguna memasukkan bilangan negatif, misal -1.
10. Tuliskan program berikut sebagai file baru (prak6-4.cpp)

```

#include <iostream>
int main()
{
    int i=0, batas;
    cout << "\nMasukkan sebuah bilangan positif : ";
    cin >> batas;
    while (i < batas)
    {
        cout << "Nilai i : " << i << endl;
        i++;
    }
    return 0;
}

```

*Listing 6.4 program menggunakan while*

11. Jalankan program tersebut secara berulang dengan nilai *batas* diubah-ubah, dan perhatikan korelasi input dan outputnya
12. Tuliskan program berikut sebagai file baru (prak6-5.cpp)

```

#include <iostream>
int main()
{
    char kata[11] = "INFORMATIKA";
    int i = 0;
    while (kata[i] != NULL)
    {
        cout << kata[i] << endl;
        i++;
    }
    return 0;
}

```

*Listing 6.5 program menggunakan while*

13. Jalankan program tersebut dan perhatikan korelasi variabel *kata* dengan output program
14. Modifikasi kode dengan mengubah isi dari variabel *kata*. Perhatikan korelasi antara variabel *kata* dengan output program

15. Tuliskan program berikut sebagai file baru (prak6-6.cpp)

```
#include <iostream>
int main()
{
    int nilai = 0;
    do{
        cout << "Total = " << nilai << endl;
        nilai++;
    }while (nilai < 5);
    cout << "Bye...";
    return 0;
}
```

*Listing 6.6 program menggunakan do-while*

16. Jalankan program tersebut dan perhatikan korelasi variabel `nilai` dengan output program

17. Modifikasi kode dengan mengubah isi dari variabel `nilai`. Perhatikan korelasi antara variabel `nilai` dengan output program

18. Tuliskan program berikut sebagai file baru (prak6-7.cpp)

```
#include <iostream>
int main()
{
    int n, i = 1;
    cout << "Masukkan bilangan positif: ";
    cin >> n;
    long total = 0;
    while (true)
    {
        if (i > n)
            break;
        total += i++;
    }
    cout << "Jumlah dari " << n << " bilangan pertama adalah " <<
total;
    return 0;
}
```

*Listing 6.7 program menggunakan while dengan break*

19. Jalankan program tersebut dan perhatikan korelasi variabel `n`, `i`, dan `total` dengan output program

20. Modifikasi program dengan mengubah nilai `n` dan perhatikan korelasi antara input dan outputnya

21. Modifikasi program dengan mengubah perhitungan penjumlahan `n` bilangan pertama menjadi perhitungan perkalian `n` bilangan pertama

22. Tuliskan program berikut sebagai file baru (prak6-8.cpp)

```
#include <iostream>
int main()
{
    int n;
    while(true)
    {
        cout << "Masukkan angka = ";
        cin >> n;
        if (n%2 == 0)
            continue;
        if (n%3 == 0)
```

```

        break;
    cout << "\tAkhir dari perulangan.\n";
    }
    cout << "\tDi luar perulangan.\n";
    return 0;
}

```

Listing 6.8 program menggunakan while *dengan break*

23. Jalankan program tersebut secara berulang dengan beberapa nilai input misalnya 4, 6, 7, 10, dll. Perhatikan korelasi input dengan output program  
Modifikasi kode dengan menambahkan atau menghilangkan statement `continue` dan `break`.

## 6.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor																
1.	CPL-03	CPMK-02	<p>Buatlah program untuk melakukan konversi suhu menggunakan perulangan, percabangan dengan ketentuan sebagai berikut:</p> <p>Tabel perhitungan suhu sebagai berikut:</p> <table><tr><td></td><td>Celcius</td><td>Kelvin</td><td>Fahrenheit</td></tr><tr><td>Celcius</td><td><math>C = C</math></td><td><math>K = C + 273</math></td><td><math>F = 9/5 * C + 32</math></td></tr><tr><td>Kelvin</td><td><math>C = K - 273</math></td><td><math>K = K</math></td><td><math>F = 9/5 * (C - 273) + 32</math></td></tr><tr><td>Fahrenheit</td><td><math>C = 5/9 * (F - 32)</math></td><td><math>K = 5/9 * (F - 32) + 273</math></td><td><math>F = F</math></td></tr></table> <p>- Pengguna memilih salah satu dari menu berikut:</p> <p>1) Konversi dari C ke F</p> <p>2) Konversi dari C ke K</p> <p>3) Konversi dari F ke C</p> <p>4) Konversi dari F ke K</p> <p>5) Konversi dari K ke C</p> <p>6) Konversi dari K ke F</p> <p>1) Keluar</p> <p>Setelah memilih menu (1-6), pengguna diminta untuk memasukkan dua buah nilai yaitu batas bawah dan batas atas suhu yang akan dikonversi. Tabel konversi suhu ditampilkan sesuai dengan pilihan menu, batas atas dan batas bawah suhu yang telah dimasukkan. Misalkan menu nomor 3 dipilih (konversi dari Fahrenheit ke Celcius), kemudian nilai batas bawah dan batas atas suhu yang dimasukkan adalah 86 dan 90 maka tabel konversi yang diitampilkan adalah hasil perhitungan konversi dari Fahrenheit ke Celcius dari suhu 86 derajat Fahrenheit sampai 90 derajat Fahrenheit. Pengguna dapat keluar dari program apabila memilih menu 0.</p> <p>- Contoh (warna merah menunjukkan nilai yang diinputkan pengguna)</p> <p>=====</p> <p>=====</p>		Celcius	Kelvin	Fahrenheit	Celcius	$C = C$	$K = C + 273$	$F = 9/5 * C + 32$	Kelvin	$C = K - 273$	$K = K$	$F = 9/5 * (C - 273) + 32$	Fahrenheit	$C = 5/9 * (F - 32)$	$K = 5/9 * (F - 32) + 273$	$F = F$	100
	Celcius	Kelvin	Fahrenheit																	
Celcius	$C = C$	$K = C + 273$	$F = 9/5 * C + 32$																	
Kelvin	$C = K - 273$	$K = K$	$F = 9/5 * (C - 273) + 32$																	
Fahrenheit	$C = 5/9 * (F - 32)$	$K = 5/9 * (F - 32) + 273$	$F = F$																	



			<div>PILIHAN MENU: 1) Konversi dari C ke F 2) Konversi dari C ke K 3) Konversi dari F ke C 4) Konversi dari F ke K 5) Konversi dari K ke C 6) Konversi dari K ke F 1) Keluar =====</div> <div>=====</div> <div>Masukkan plihan menu = 3 Masukkan batas bawah dan batas atas suhu yang ingin dikonversi = 86 90</div> <table><thead><tr><th>F</th><th>C</th></tr></thead><tbody><tr><td>---</td><td>---</td></tr><tr><td>86</td><td>30</td></tr><tr><td>87</td><td>30.56</td></tr><tr><td>88</td><td>31.11</td></tr><tr><td>89</td><td>31.67</td></tr><tr><td>90</td><td>32.22</td></tr></tbody></table> <div>=====</div> <div>=====</div> <div>PILIHAN MENU: 1) Konversi dari C ke F 2) Konversi dari C ke K 3) Konversi dari F ke C 4) Konversi dari F ke K 5) Konversi dari K ke C 6) Konversi dari K ke F 1) Keluar =====</div> <div>=====</div> <div>Masukkan plihan menu = 0 Keluar dari program...</div>	F	C	---	---	86	30	87	30.56	88	31.11	89	31.67	90	32.22	
F	C																	
---	---																	
86	30																	
87	30.56																	
88	31.11																	
89	31.67																	
90	32.22																	

## 6.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-02	20%		
2.	Praktik	CPL-03	CPMK-02	50%		
3.	Post-Test	CPL-03	CPMK-02	30%		
Total Nilai						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 7: PERULANGAN BERTINGKAT

Pertemuan ke : 7

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-02	Mampu mengimplementasi program terstruktur sesuai dengan spesifikasi program menggunakan perintah eksekusi bahasa pemrograman

### 7.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep perulangan bertingkat dengan pernyataan `for`, `while`, `do while` dan kombinasi yang diperlukan dalam membuat perulangan bertingkat.
2. Menerapkan konsep perulangan bertingkat dengan `for`, `while`, `do while`, `do` dan kombinasinya kedalam sebuah program C++.

### 7.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-02	Kemampuan mahasiswa dalam menjelaskan dan menerapkan konsep perulangan bertingkat dengan <code>for</code> , <code>while</code> , <code>do while</code> , <code>do</code> dan kombinasinya kedalam sebuah program C++.
--------	---------	---

### 7.3 TEORI PENDUKUNG

Dalam implementasi pemrograman banyak masalah yang harus diselesaikan secara berulang bertingkat (*nested loop*). Nested loop dapat diimplementasi baik menggunakan `for` maupun menggunakan `while`. Terdapat empat kemungkinan nested loop yang menggunakan `while` dan `for`

**a. For dalam for**, bentuk umum statement ini adalah :

```
for (expression1; expression2; expression3)
for (expression4; expression5; expression6)
```

```
statement;
```

**b. While dalam while, bentuk umum statement ini adalah:**

```
while (condition1)
while (condition 2)
    statement;
```

**c. For dalam while, bentuk umum statement ini adalah:**

```
while (condition1)
for (expression1; expression2; expression3)
    statement;
```

**d. While dalam for, bentuk umum statement ini adalah:**

```
for (expression1; expression2; expression3)
while (condition1)
    statement;
```

## 7.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

8 Komputer.

9 Dev C++.

## 7.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	Jelaskan apa yang dimaksud dengan perulangan bertingkat.	30
2.	CPL-03	CPMK-02	Tuliskan program tulis membuat output berikut dengan perulangan bertingkat: 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5	70

## 7.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-01	Selesaikan langkah praktikum 1 – 6	Hasil praktikum langkah 1 – 6	30
2.	CPL-03	CPMK-01	Selesaikan langkah praktikum 7	Hasil praktikum langkah 7	30
3	CPL-03	CPMK-01	Selesaikan langkah praktikum 8 – 9	Hasil praktikum langkah 8 – 9	40

### Langkah-Langkah Praktikum:

1. Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
2. Tuliskan program berikut sebagai file baru (prak7-1.cpp) dan perhatikan apa output dari program berikut ini

```
#include <iostream>
using namespace std;

int main ()
{
    int i,j;
    int n = 5;
    for (i=1; i<=n; i++)
    {
        for (j=1; j<=i; j++)
        {
            cout<<1;
        }
        cout<<endl;
    }
}
```

*Listing 7.1 program menggunakan for dalam for*

3. Apakah hasil output dari program di atas?
4. Ubahlah kode `cout<<1;` menjadi `cout<<j;`
5. Apakah hasil output dari program yang telah dimodifikasi?
6. Buatlah program yang menghasilkan output sebagai berikut.

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

7. Tuliskan program berikut sebagai file baru (prak7-2.cpp) dan perhatikan apa output dari program berikut ini

```
#include <iostream>
using namespace std;

int main()
{
    int i,j,n;
    cout << " Tampilan segitiga "<<endl;
    cout << " Jumlah baris: ";
    cin >> n;
    for(i=0;i<n;i++)
    {
```

```

        for (j=1; j<=n-i; j++)
            cout<<" ";
        for (j=1; j<=2*i-1; j++)
            cout<<"*";
        cout<<endl;
    }
}

```

*Listing 7.2 program menampilkan segitiga*

8. Buatlah program untuk menampilkan output berikut ini dan simpan dengan nama `prak7-3`.

Output :

```

1
121
12321
1234321
123454321

```

9. Lakukan modifikasi kode program 7-3 menjadi output berikut ini.

Output :

```

1
121
12321
1234321
123454321
1234321
12321
121
1

```

## 7.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-02	<p>Buatlah program yang memberikan tampilan berikut ini dengan perulangan:</p> <pre> 6 5 6 4 5 6 3 4 5 6 2 3 4 5 6 1 2 3 4 5 6 </pre>	50
2.	CPL-03	CPMK-02	<p>Buatlah program untuk menampilkan perkalian kolom dan baris menggunakan perulangan dengan tampilan output berikut ini:</p> <pre> 1 2 3 4 5 6 2 4 6 8 10 12 3 6 9 12 15 18 4 8 12 16 20 24 </pre>	50

## 7.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-02	20%		
2.	Praktik	CPL-03	CPMK-02	50%		
3.	Post-Test	CPL-03	CPMK-02	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--



## PRAKTIKUM 8: FUNGSI

**Pertemuan ke** : 8

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-03	Mampu mengimplementasi solusi masalah menjadi subrutin, menuliskan subrutin dengan kode bahasa pemrograman, menggunakan ulang (re-use) subrutin pada pemrograman

### 8.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep tentang fungsi (fungsi library dan fungsi user defined), struktur fungsi, dan parameter pada fungsi.
2. Menerapkan konsep pembuatan fungsi dan pemanggilanya dengan pass by value dan pass by reference pada sebuah program C++.

### 8.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-03	Kemampuan mahasiswa dalam menerapkan penggunaan fungsi library dan pembuatan fungsi user defined pada bahasa pemrograman C++.
--------	---------	---

### 8.3 TEORI PENDUKUNG

Fungsi merupakan blok kode program atau sekelompok kode program yang digunakan untuk mengerjakan tugas khusus. Fungsi dapat diletakkan dalam library/file header sehingga jika program lain akan menggunakannya, cukup menyertakan file header tersebut dengan statement *include* pada awal program.

Secara umum, berdasarkan jenisnya fungsi dapat dibedakan menjadi dua kategori:

**Predefined Functions** (fungsi standart): merupakan fungsi dasar yang telah disediakan oleh c++ dan pemrogram tinggal menggunakannya. Contohnya strlen(), getch(), dll

**User-Defined Functions:** Fungsi yang dibuat oleh programmer sendiri sesuai dengan kebutuhan program.

Selain kategori tersebut, fungsi juga dibedakan menjadi 2 berdasarkan fungsinya, yaitu fungsi yang tidak memiliki nilai kembalian atau sering disebut dengan prosedur dan fungsi yang memiliki nilai kembalian atau disebut dengan fungsi. Fungsi yang tidak memiliki nilai kembalian maka cara pembuatannya diawali dengan statmen void sedangkan fungsi yang memiliki nilai kembalian cara pembuatannya diawali dengan statemen tipe data dari nilai kembalian tersebut.

Fungsi tanpa nilai kembalian ditulis dengan struktur sebagai berikut:

```
Void NamaFunction (tipe data variable1, tipe data variable2, tipe data variable3,...)
{
    deklarasi variable lokal;
    statements;
}
```

Sedangkan fungsi dengan nilai kembalian ditulis dengan struktur sebagai berikut:

```
tipe data kembalian NamaFunction (tipe data variable1,
tipe data variable2, tipe data variable3, ...)
{
    deklarasi variable lokal;
    statements;
    return tipe data kembalian;
}
```

*Tipe data kembalian merupakan / menunjukan tipe data yang akan dikembalikan ketika fungsi tsb akan dipanggil. Hanya ada satu nilai kembalian dari function.*

## 8.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

9 Komputer.

10 Dev C++

## 8.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Jelaskan apa yang dimaksud dengan predefine function atau fungsi standart, berikan contoh sebuah fungsi dan kegunaannya serta file header (library) yang di gunakan.	50
2.	CPL-03	CPMK-03	Jelaskan perbedaan mendasar fungsi yang memiliki nilai kembalian dan fungsi yang tidak memiliki nilai kembalian	50

## 8.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-03	Selesaikan langkah praktikum 1 – 4	Hasil praktikum langkah 1 – 4	30

2.	CPL-03	CPMK-03	Selesaikan langkah praktikum 5 – 12	Hasil praktikum langkah 5 – 12	40
3.	CPL-03	CPMK-03	Selesaikan langkah praktikum 13 – 18	Hasil praktikum langkah 13 – 18	30

### Langkah-Langkah Praktikum:

1. Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
2. Tuliskan kode berikut ini dan simpan dalam file `Prak8-1.cpp`. Kode berikut ini memanggil library `string.h` dan menggunakan fungsi dari library `string.h` yaitu fungsi `strcpy`.

```

/* File program : prak8-1.cpp */

#include <iostream>
#include <string.h>
using namespace std;
int main() {
    char info[30];
    strcpy(info, "TEKNIK INFORMATIKA FTI UAD");
    /*fungsi dari string.h*/
    cout<<info <<endl;
    cout<<"panjang karakter=" << strlen(info);
    /*fungsi dari string.h*/
    cout<<"\n";
    return 0;
}

```

3. Jalankan program tersebut secara berulang dan gantilah pernyataan “TEKNIK INFORMATIKA FTI UAD” dengan tulisan yang lain. Amati setiap perubahan program dan outputnya .
4. Tuliskan kode berikut ini dan simpan dalam file `Prak8-2.cpp`. Kode berikut ini memanggil library `math.h` dan menggunakan fungsi dari library `math.h` yaitu fungsi `fmax`, `fmin`, `pow`.

```

/* File program : prak8-2.cpp */
#include <iostream>
#include <math.h>

using namespace std;
int main()
{
    int a,b;
    cout<<"Nilai A="; cin>>a;
    cout<<"Nilai B="; cin>>b;
    cout<<"nilai terbesar antara A dan B="<< fmax(a,b)
    <<endl;
    cout<<"nilai terkecil antara A dan B="<< fmin(a,b)
    <<endl;
    cout<<"A pangkat B ("<<a <<"^"<<b<<")="<<
    pow(a,b)<<endl;
    return 0;
}

```

5. Ketiklah kode program di bawah ini dan simpan dalam file prak8-3.cpp. Kode berikut ini mengandung fungsi `cetakgaris` yang tidak memiliki kembalian.

```
/* File program : prak8-3.cpp */
#include<iostream>
#include<conio.h>
    using namespace std;

/* fungsi untuk mencetak garis dgn panjang n karakter */
void cetakgaris(int n)
{ int i;
  for(i=0;i<=n;i++)
    cout<<"-";cout<<"\n";
}
main()
{
  cetakgaris(15); //pemanggilan fungsi
  getch();
}
```

6. Jalankan program tersebut secara berulang dengan nilai variabel yang diubah-ubah, atau dengan cara pemanggilan fungsi yang berubah dan analisislah korelasi input dan outputnya.
7. Ketik program di bawah ini dan disimpan dalam file prak8-4.cpp. Kode berikut ini mengandung beberapa fungsi dengan input dan output.

```
/* File program : prak8-4.cpp */
#include<iostream>
#include<conio.h>
    using namespace std;
int Luas(int x,int y){
    int hasil;
    return hasil =x*y;
}
float KLL(int r){
    float keling,phi=22/7;
    keling=2*phi*r;
    return keling;
}
void cetakgaris(int n){
    int i;
    cout<<"\n\n";
    for(i=0;i<=n;i++)
        cout<<"-";cout<<"\n";
}
int main(){
    int p=0,l=0;
    float K;
    cout<<"Menghitung Luas Persegi 4"<<endl;
    cout<<endl<<"Masukan Panjang : ";cin>>p;
    cout<<endl<<"Masukan Lebar : ";cin>>l;
    cout<<endl<<"Luas = "<< Luas(p,l); //pemanggilan fungsi
    cetakgaris(20); //pemanggilan fungsi
}
```

```

        cout<<"\n\n";
        cout<<"Menghitung Keliling lingkaran"<<endl;
        cout<<"Masukan jari : ";
        cin>>l;
        K=KLL(l); //pemanggilan fungsi
        cout<<"Keliling Lingkaran = " <<K;
        getch();
    }

```

8. Jalankan program di atas, masukkan nilai input yang berbeda-beda, lihatlah perubahan pada outputnya. Analisis lah hasil yang dikembalikan oleh setiap fungsi ketika diberikan suatu nilai input.
9. Ketik program di bawah ini dan simpan dalam file prak8-5.cpp. Program berikut ini mengandung fungsi dengan input "pass by reference". Apa tujuan fungsi dengan input "pass by reference"? Untuk mengubah nilai pada variabel input dan perubahannya persisten (tersimpan/tertulis) dalam memori.

```

/* File program : prak8-5.cpp */
#include<iostream>
#include<conio.h>
using namespace std;
void tambah2(int &a){
    a=a+2;
}
main(){
    int x;
    cout<<"Nilai X ="; cin >>x;

    cout<<"Nilai awal x : " <<x;
    tambah2(x);
    cout<<"Nilai x setelah pemanggilan fungsi =" << x;
    getch();
}

```

10. Jalankan program tersebut dan amati hasil yang dikembalikan.
11. Ketik program di bawah ini dan simpan dalam file prak8-6.cpp.

```

/* File program : prak8-6.cpp */
#include<iostream>
#include<conio.h>
using namespace std;
int pangkat(int a, int b) {
    int hasil,i;
    hasil=1;
    for (i=1; i<=b; i++)
        hasil*=a;
    return hasil;
}
main (){
    int x,y,hsl;
    cout<<"Nilai X="; cin>>x;
    cout<<"Nilai Y="; cin>>y;
}

```

```

    hsl=pangkat(x,y);
    cout<<x <<" Pangkat " << y <<"=" <<hsl;
}

```

12. Jalankan program tersebut, ubah nilai X dan nilai Y dengan beberapa nilai yang berbeda. Perhatikan output programnya.
13. Sekarang Anda akan membuat sebuah library. Pertama-tama buatlah *New Project* di dalam *Dev C++* dengan cara pilih menu *File → New → Project*. Kemudian Anda akan diminta untuk memilih jenis project yang akan dibuat. Pilihlah *Console Application* lalu beri nama Project Anda dengan nama *LibraryProject*. Setelah itu akan otomatis terbentuk file *main.cpp*.
14. Sekarang kita akan membuat sebuah library dalam file *lingkaran.h*

Buatlah file baru melalui menu *Project → New File* kemudian simpan ke dalam file *lingkaran.h*  
 Ketik kode berikut dalam file *lingkaran.h*

```

#ifndef _LINGKARAN_H
#define _LINGKARAN_H
float HitungKeliling(int r);
float HitungLuas(int r);
#endif

```

15. Sekarang, buatlah file *lingkaran.c* dengan cara yang sama *Project → New File*  
 Ketik kode berikut dalam file *lingkaran.c* (kode berikut adalah implementasi dari fungsi *HitungKeliling* dan *HitungLuas* dari file *lingkaran.h*)

```

#include "lingkaran.h"
float HitungKeliling(int r)
{
    return (3.14*2*r);
}
float HitungLuas(int r)
{
    return (3.14*r*r);
}

```

16. Sebuah library telah Anda buat. Sekarang Anda akan panggil library tersebut dari program utama (*main.cpp*)

```
#include <iostream>
#include "lingkaran.h"
using namespace std;

int main(int argc, char** argv) {
    int jari=10.0;
    cout<<HitungKeliling(jari)<<endl;
    cout<<HitungLuas(jari)<<endl;
    return 0;
}
```

17. Agar project yang terdiri dari file library dan file program utama dapat dijalankan, maka Anda perlu melakukan build project tersebut. Caranya pilih *Execute* → *Rebuild All*. Pada saat Anda melakukan *Rebuild All* maka Anda melakukan dua hal berikut ini:

- (1). Kompilasi semua file bertipe C/C++
- (2). Linking, yaitu proses menghubungkan antara file program utama dengan file-file library yang dipanggil dari program utama

Perhatikan hasil *Rebuild All* berikut ini.

```
Rebuilding whole project...
-----
- Project Filename: C:\Users\Dewi Ismi\Documents\Lingkaran.dev
- Compiler Name: TDM-GCC 4.9.2 64-bit Release

Building makefile...
-----
- Filename: C:\Users\Dewi Ismi\Documents\Makefile.win

Processing makefile...
-----
- Makefile Processor: C:\Program Files (x86)\Dev-Cpp\MinGW64\bin\mingw32-make.exe
- Command: mingw32-make.exe -f "C:\Users\Dewi Ismi\Documents\Makefile.win" clean all

rm.exe -f main.o lingkaran.o Lingkaran.exe
g++.exe -c main.cpp -o main.o -I"C:\Program Files (x86)\Dev-Cpp\MinGW64\include" -I"C:\Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-mingw32\include"
g++.exe -c lingkaran.c -o lingkaran.o -I"C:\Program Files (x86)\Dev-Cpp\MinGW64\include" -I"C:\Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-mingw32\include"
g++.exe main.o lingkaran.o -o Lingkaran.exe -L"C:\Program Files (x86)\Dev-Cpp\MinGW64\lib" -L"C:\Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-mingw32\lib"

Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Dewi Ismi\Documents\Lingkaran.exe
- Output Size: 1,832,935,332,5195 MiB
- Compilation Time: 0,88s
```

18. Setelah *Rebuild All* telah dijalankan, sekarang Anda dapat menjalankan program Anda, *Execute* → *Run*. Lihat apa output dari program Anda, perhatikan hasilnya.

## 8.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Buatlah sebuah program kalkulator yang berisi operasi penambahan, pengurangan, perkalian, pembagian dua buah bilangan. Contoh tampilan program adalah sebagai berikut ini dengan warna merah adalah input dari user.	

			PROGRAM KALKULATOR Menu Operasi: 1. Penambahan 2. Pengurangan 3. Perkalian 4. Pembagian Pilih Menu No: 2 Bilangan Pertama: 10 Bilangan Kedua: 5 Hasil 10 - 5 = 5  Buatlah program dengan fungsi untuk masing operasi dalam satu file cpp  Buatlah program dalam aplikasi console dengan fungsi yang tersimpan pada ibarary	30  40
2.	CPL-03	CPMK-03	Buatlah analisa dan penjelasan dari program diatas	30

## 8.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-03	20%		
2.	Praktik	CPL-03	CPMK-03	50%		
3.	Post-Test	CPL-03	CPMK-03	30%		
Total Nilai						



**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 9: FUNGSI REKURSIF

**Pertemuan ke** : 9

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-03	Mampu mengimplementasi solusi masalah menjadi subrutin, menuliskan subrutin dengan kode bahasa pemrograman, menggunakan ulang (re-use) subrutin pada pemrograman

### 9.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep fungsi rekursif dan karakteristiknya.
2. Menerapkan fungsi rekursif kedalam sebuah program komputer dengan C++

### 9.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-03	Kemampuan mahasiswa dalam menjelaskan dan menerapkan fungsi rekursif kedalam sebuah program C++.
--------	---------	--

### 9.3 TEORI PENDUKUNG

Fungsi rekursif adalah sebuah fungsi yang didalamnya terdapat perintah untuk memanggil dirinya sendiri atau sebuah fungsi yang memanggil fungsi yang lain, dan fungsi lain tersebut terdapat perintah yang memanggil fungsi dirinya sendiri. Sebagai contoh sebuah fungsi  $f()$  disebut sebagai fungsi rekursif apabila memanggil lain misal fungsi  $g()$  dan didalam fungsi  $G$  tersebut terdapat perintah yang memanggil fungsi  $f()$ .

Pada fungsi biasa, proses berulang biasanya dinyatakan dengan iterative yang dinyatakan dengan perintah perulangan seperti for atau while. Fungsi rekursif dapat dipandang sebagai sebuah operator. Sebagai contoh dalam kasus perhitungan nilai faktorial secara iteratif  $n$  faktorial didefinisikan sebagai:

$$n! = 1 \times 2 \times 3 \times 4 \times \dots \times N = \prod_{i=1}^n i$$

n faktorial secara rekursif dapat didefinisikan sebagai berikut:

$$n! = n * (n-1)!$$

$$(n-1)! = (n-1) * (n-2)!$$

$$(n-2)! = (n-2) * (n-3)!$$

....

$$1! = 1$$

$$0! = 1$$

## 9.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

10 Komputer.

11 Dev C++

## 9.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Jelaskan apa yang dimaksud dengan fungsi rekursif	30
2.	CPL-03	CPMK-03	Bagaimana karakteristik fungsi rekursif	40
3	CPL-03	CPMK-03	Jelaskan apa yang dimaksud dengan <i>stopping poin</i> pada fungsi rekursif,	30

## 9.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-03	Selesaikan langkah praktikum 1 – 5	Hasil praktikum langkah 1 – 5	35
2.	CPL-03	CPMK-03	Selesaikan langkah praktikum 6 – 8	Hasil praktikum langkah 6 – 8	35
	CPL-03	CPMK-03	Selesaikan langkah praktikum 9 – 10	Hasil praktikum langkah 9 – 10	30

**Langkah-Langkah Praktikum:**

- Ikuti langkah-langkah di bawah ini, ketik sendiri dan di larang COPY-PASTE.
- Ketik kode program berikut ini dan simpan dengan nama prak9-1.cpp. Berikut adalah program untuk menjumlahkan angka bilangan bulat dari 1 sampai dengan N.

```
/* File program : prak9-1.cpp */
#include<iostream>
```

```
#include<conio.h>
using namespace std;
int jumlah_deret(int a) {
if (a==0){
    return a;
}
else {
    return a + jumlah_deret (a-1);
}
}
main (){
    int n,hsl;
    cout<<"Nilai N="; cin>>n;
    hsl=jumlah_deret(n);
    cout<< "Jumlah Deret " << n <<"=" << hsl;
}
```

3. Jalankan program tersebut beberapa kali dengan memberikan input nilai n yang berbeda-beda kemudian amati hasilnya.
4. Ketik kode program berikut ini kemudian simpan dengan nama file prak9-2.cpp. Program berikut ini adalah program untuk menghitung nilai factorial tertentu (N).

```
/* File program : prak9-2.cpp */
#include <iostream>
using namespace std;

int faktorial(int n);

int main(){
    int n;
    cout<<"Masukan N="; cin>>n;
    cout<<"Nilai factorial "<< n<<" adalah = "
<<faktorial(n)<<endl;
    return 0;
}

int faktorial(int n){
    int nilai=n;
    if (n==1)
        return nilai;
    else {
        return nilai*faktorial(n-1);
    }
}
```

5. Jalankan program tersebut beberapa kali dengan memberikan nilai input N yang berbeda-beda kemudian amati hasilnya.
6. Ketik program di bawah ini dan simpan dalam file prak9-3.cpp. Program di bawah ini adalah fungsi rekursif untuk menghitung pangkat dari suatu bilangan.

```

/* File program : prak9-3.cpp */
#include<iostream>
#include<conio.h>
using namespace std;
int pangkat(int a, int b) {
    if (b==1){
        return a;
    }
    else {
        return a * pangkat (a,b-1);
    }
}
main (){
    int x,y,hsl;
    cout<<"Nilai X="; cin>>x;
    cout<<"Nilai Y="; cin>>y;
    hsl=pangkat(x,y);
    cout<<x <<" Pangkat " << y <<"=" <<hsl;
}

```

7. Jalankan program tersebut, ubah nilai X dan nilai Y dengan beberapa nilai yang berbeda. Perhatikan output programnya.
8. Ketik program di bawah ini dan simpan dalam file prak9-4.cpp. Program di bawah ini adalah fungsi rekursif untuk menghitung jumlah deret bilangan genap dari 0 sampai dengan n.

```

/* File program : prak9-4.cpp */
#include<conio.h>
using namespace std;
int jumlah_deret_genap(int a) {
    if (a==0){
        return a;
    }
    else {
        if (a%2==0)
            return a + jumlah_deret_genap (a-1);
        else
            return jumlah_deret_genap(a-1);
    }
}
main (){
    int n,hsl;
    cout<<"Nilai N="; cin>>n;
    hsl=jumlah_deret_genap(n);
    cout<< "Jumlah Deret Bilangan Genap " << n <<"=" << hsl;
}

```

9. Jalankan program tersebut beberapa kali dan ubah nilai n dengan beberapa nilai yang berbeda. Perhatikan output programnya.
10. Modifikasi program prak9-4.cpp menjadi program untuk menghitung deret bilangan ganjil dari 0 sampai dengan n.

## 9.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor																																	
1.	CPL-03	CPMK-03	<p>Buatlah sebuah program untuk menghitung jumlah akumulasi jumlah pendapatan dalam waktu 10 bulan. Contoh data pendapatan seperti pada tabel berikut:</p> <table><tr><th>Bulan ke</th><th>Pendapatan</th><th>Jumlah</th></tr><tr><td>1</td><td>100000</td><td>100000</td></tr><tr><td>2</td><td>200000</td><td>300000</td></tr><tr><td>3</td><td>75000</td><td>375000</td></tr><tr><td>4</td><td>100000</td><td>475000</td></tr><tr><td>5</td><td>125000</td><td>600000</td></tr><tr><td>6</td><td>150000</td><td>750000</td></tr><tr><td>7</td><td>150000</td><td>900000</td></tr><tr><td>8</td><td>200000</td><td>1100000</td></tr><tr><td>9</td><td>250000</td><td>1350000</td></tr><tr><td>10</td><td>150000</td><td>1500000</td></tr></table> <p>Jumlah pendapatan tiap bulanya (tanda merah) dapat merupakan input dari user.</p>	Bulan ke	Pendapatan	Jumlah	1	100000	100000	2	200000	300000	3	75000	375000	4	100000	475000	5	125000	600000	6	150000	750000	7	150000	900000	8	200000	1100000	9	250000	1350000	10	150000	1500000	70
Bulan ke	Pendapatan	Jumlah																																			
1	100000	100000																																			
2	200000	300000																																			
3	75000	375000																																			
4	100000	475000																																			
5	125000	600000																																			
6	150000	750000																																			
7	150000	900000																																			
8	200000	1100000																																			
9	250000	1350000																																			
10	150000	1500000																																			
2.	CPL-03	CPMK-03	Tuiskan analisa dan penjelasan untuk program diatas	30																																	

## 9.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-03	20%		
2.	Praktik	CPL-03	CPMK-03	50%		
3.	Post-Test	CPL-03	CPMK-03	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 10:ARRAY

Pertemuan ke : 10

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-03	Mampu mengimplementasi solusi masalah menjadi subrutin, menuliskan subrutin dengan kode bahasa pemrograman, menggunakan ulang (re-use) subrutin pada pemrograman

### 10.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

3. Menjelaskan konsep array 1 dimensi yang meliputi deklarasi, pengisian nilai dan pembacaan nilai array.
4. Menerapkan konsep array 1 dimensi pada program komputer dengan C++

### 10.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-03	Kemampuan mahasiswa dalam menjelaskan dan menerapkan konsep array 1 dimensi dan penggunaanya dalam membuat program dengan C++.
--------	---------	--

### 10.3 TEORI PENDUKUNG

Array merupakan kelompok lokasi memory yg dihubungkan secara nyata bahwa semua memiliki nama dan tipe data yg sama. Compiler harus menyediakan/ mengalokasi tempat untuk setiap element dari array yg dideklarasikan. Ukuran dari array statis (fixed) selama eksekusi program. Untuk merujuk lokasi bagian/element array digunakan index/sub scrip dan posisi nomor dari bagian/element array tersebut. Array dideklarasikan spesifikasi **tipe data**, yang memuat **nama** dan **ukuran**, yg mana komputer akan menyediakan sejumlah memory sesuai yg dideklarasikan.

Syntax umum penulisan array:



`data_type array_name[size];`

Alamat / nomor index array dimulai dari 0

Contoh deklarasi array:

```
int nilai[100];
char nama[20];
double bigval[5*200];
int a[27], b[10], c[76];
```

Ada 2 cara untuk menginisialisasi nilai array : saat kompilasi dan saat eksekusi

Contoh deklarasi saat kompilasi dengan nama variable `arr` :

- `int arr[] = {1, 2, 3, 4, 5};` → array tak berukuran

Kita dapat mendefinisikan berapa elemen yg kita butuhkan tanpa memberikan ukurannya. Kita juga dapat mendefinisikan hanya 3 elemen dengan ukuran diberikan

- `int arr[3] = {90, 21, 22};`

Inisialisasi 2 elemen pertama dgn nilai 2 dan 4, sedangkan elemen lain diinisialisasikan dengan nol

- `int arr[5] = {2, 4};`

Inisialisasi semua elemen array dengan nol

`int arr[5] = {0};`

## 10.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

11 Komputer.

12 Dev C++.

## 10.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Jelaskan apa yang dimaksud dengan array	30
2.	CPL-03	CPMK-03	Tuliskan sintak untuk mendeklarasikan array 1 dimensi dengan jumlah elemen 5 dengan tipe integer, mengisi array elemen ke 2 dengan nilai 4, dan mencetak nilai elemen ke 2 dengan.	70

## 10.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-03	Selesaikan langkah praktikum 1 – 3	Hasil praktikum langkah 1 – 3	20
2.	CPL-03	CPMK-03	Selesaikan langkah praktikum 4 – 7	Hasil praktikum langkah 4 – 7	40
3	CPL-03	CPMK-03	Selesaikan langkah praktikum 8 – 11	Hasil praktikum langkah 8 – 11	40

**Langkah-Langkah Praktikum:**

1. Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
2. Ketik program di bawah ini dan simpan dalam file prak10-1.cpp. Program di bawah ini mengandung array of integer dan menampilkan elemen array pada konsole serta menjumlahkan total elemen array.

```
/* File program prak10-1.cpp */
#include<iostream>
#include<conio.h>
using namespace std;

int main()
{
    int i, total, jum_mhs[7]={40, 39, 41, 36, 38, 34, 42};
    total=0;
    for(i=1; i<7; i++)
    {
        cout<<"Jumlah Mahasiswa Kelas "<<i<<": "<<jum_mhs[i]<<endl;
        total+=jum_mhs[i];
    }
    cout<<"\nJumlah Total Mahasiswa = "<<total;
    getch();
}
```

3. Jalankan porgam tersebut secara berulang dengan nilai variable yang diubah-ubah, atau dengan cara pemanggilan fungsi yang berubah. Kemudian lakukan analisa hubungan antara input dan output.
4. Tuliskan program berikut sebagai file baru dalam file prak10-2.cpp.

```
/* File program prak10-2.cpp*/
#include<iostream>
#include<conio.h>
using namespace std;

int main()
{
    int i, cari, data[10]={2, 3, 6, 5, 7, 8, 10, 9, 1, 4};
    cout<<"Masukkan nilai yang ingin dicari: ";
    cin>>cari;
    for (i=0; i<10; i++)
    {
        if (data[i]==cari)
        {cout<<"\nData yang anda cari berada pada indeks ke- "<<i;
        break;
        }
    }
    if (i==10)
    cout<<"Data yang anda cari tidak ditemukan";
    getch();
}
```

5. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.

6. Ketik program di bawah ini dan simpan dengan nama file prak10-3.cpp. Program ini merupakan program untuk menghitung penjumlahan dari dua buah deret dengan elemen array yang sama besar.

```
/* File program prak10-3.cpp */
#include <iostream>
using namespace std;
main(){
    int A[5];
    int B[5];
    int C[5];
    cout<<"Program hitung penjumlaha 2 buah deret" <<endl;
    cout<<"Deret A" <<endl;
    for (int i=0; i<5; i++){
        cout<<"Data A[" <<i<< "]="; cin>>A[i];
    }
    cout<<"Deret B" <<endl;
    for (int i=0; i<5; i++){
        cout<<"Data B[" <<i<< "]="; cin>>B[i];
    }
    for (int i=0; i<5; i++){
        C[i]=A[i]+B[i];
    }
    cout<<"Hasil penjumlaha deret" <<endl;
    for (int i=0; i<5; i++){
        cout<<"C[" <<i<< "]= A["<<i<<" + B["<<i<<"=" <<A[i] <<"
+ " <<B[i] <<"=" << C[i] <<endl;
    }
}
```

7. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.
8. Ketik program di bawah ini dan simpan dengan nama file prak10.4. Program ini adalah contoh array untuk menyimpan data karakter.

```
/* File program prak10.4 */
#include <iostream>
using namespace std;
main(){
    char nama[4]={'a','g','u','s'};
    string alamat[5]={"Dukuh Sukahati","Kelurahan Damai","Kecamatan
Tenangraya","Kabupaten Gunungkidul","Propinsi DIY"};

    cout <<"Nama=";
    for (int i=0; i<4; i++){
        cout<<nama[i];
    }
    cout<<endl;
    cout<<"Alamat:" <<endl;
    for (int i=0; i<5; i++){
        cout<<alamat[i]<<endl;
    }
}
```

9. Jalankan program tersebut dan amati outputnya.
10. Ketik program di bawah ini dan simpan dengan nama file prak10.5, program ini adalah contoh pemanfaatan pemanfaatan array untuk mencari karakter pada suatu kalimat.

```

/* File program prak10.5 */
#include <iostream>
#include <string.h>
#include <conio.h>
using namespace std;

int main()
{
    char cari, kalimat[100];
    int a,tot=0;
    cout<<"Program Mencari Data Pada Kalimat\n";
    cout<<"=====\n";
    cout<<"Masukkan Kalimat : ";
    gets(kalimat);
    cout<<"-----\n";
    a=strlen(kalimat);
    //input karakter
    cout<<"\nMasukkan data yang ingin anda cari : ";cin>>cari;
    cout<<"-----\n";

    for(int i=0;i<a;i++)
    {
        if(kalimat[i]==cari)
        {
            cout<<"Data "<<cari<<" ditemukan pada urutan ke-
"<<i+1<<endl;
            tot+=1;
        }
    }
    cout<<"=====\n";
    if(tot!=0)
    {
        cout<<"Data "<<cari<<" berjumlah "<<tot;
    }
    else
    {
        cout<<"Data tidak ditemukan!!!";
    }
    getch();
}

```

11. Jalankan program berulang kali kemudian masukan kalimat dan pencarian yang berbeda-beda. Amati setiap hasil output program.

## 10.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Buatlah sebuah program sederhana untuk mencari dan menampilkan bilangan-bilangan ganjil atau genap dari elemen array[10]. Nilai array merupakan input dari user.	70

2.	CPL-03	CPMK-03	Tuliskan analisa dan penjelasan dari program diatas	30
----	--------	---------	---	----

## 10.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-03	20%		
2.	Praktik	CPL-03	CPMK-03	50%		
3.	Post-Test	CPL-03	CPMK-03	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 11:ARRAY 2 DIMENSI

**Pertemuan ke** : 11

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-03	Mampu mengimplementasi solusi masalah menjadi subrutin, menuliskan subrutin dengan kode bahasa pemrograman, menggunakan ulang (re-use) subrutin pada pemrograman

### 11.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

5. Menjelaskan konsep array dua dimensi meliputi deklarasi, pengisian nilai, pembacaan nilai array dan penggunaannya
6. Menerapkan konsep array dua dimensi kedalam sebuah program komputer dengan C++.

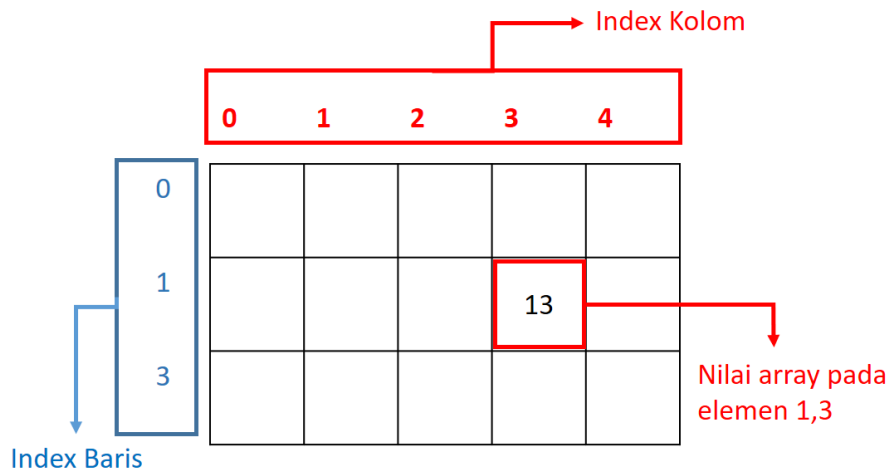
### 11.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-03	Kemampuan mahasiswa dalam menjelaskan dan menerapkan konsep array dua dimensi yang meliputi deklarasi, pengisian nilai dan pembacaan nilai array kedalam sebuah program komputer dengan C++.
--------	---------	--

### 11.3 TEORI PENDUKUNG

Array 2 dimensi adalah array yang memiliki 2 dimensi atau 2 indeks: baris dan kolom. Sama seperti array 1 dimensi, nilai array pada array 2 dimensi harus memiliki tipe data yang sama. Array 2 dimensi dapat dikatakan berbentuk seperti matriks atau tabel. Gambar berikut menjelaskan ilustrasi dari array 2 dimensi.



Gambar 11.1 Ilustrasi Array 2 Dimensi

Dari gambar di atas dapat kita lihat bahwa array a merupakan array 2 dimensi yang merupakan sebuah matriks dengan ordo 3x5. Contoh array di atas memiliki 3 elemen array dimana masing-masing elemennya berupa array dengan 5 elemen, atau bisa kita katakan memiliki 3 baris dan 5 kolom. Oleh karena itu, array multidimensi juga sering disebut dengan “array dari array” atau “kumpulan array”.

#### 11.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

- 12 Komputer.
- 13 Dev C++.

#### 11.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-01	Jelaskan apa yang dimaksud dengan array 2 dimensi	30
2.	CPL-03	CPMK-01	Tuliskan sintak untuk mendeklarasikan array 2 dimensi dengan jumlah elemen baris 3 dan kolom 5 dengan tipe integer, mengisi array elemen baris ke 2 dan kolom ke 3 dengan nilai 34, dan mencetak nilai array pada elemen baris ke 2 kolom ke 3.	70

#### 11.6 LANGKAH PRAKTIKUM

Aturan Penilaian (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-03	Selesaikan langkah praktikum 1 – 4	Hasil praktikum langkah 1 – 4	40
2.	CPL-03	CPMK-03	Selesaikan langkah praktikum 5 – 7	Hasil praktikum langkah 5 – 7	40
3	CPL-03	CPMK-03	Selesaikan langkah praktikum 8– 9	Hasil praktikum langkah 8 – 9	20



### Langkah-Langkah Praktikum:

1. Ikuti langkah-langkah berikut ini dengan mengetik sendiri kode program, dilarang melakukan COPY PASTE.
2. Ketik kode program berikut ini kemudian simpan file dengan nama prak11-1.cpp.

```

/* File program prak11-1.cpp */
#include<iostream>
#include<conio.h>
using namespace std;

int main()
{
    int i, j, hasil[4][2];
    int data[4][2]={5, 6, 3, 2, 6, 8, 9, 11};
    int data2[4][2]={2, 3, 1, 5, 6, 8, 4, 9};

    //menampilkan isi matriks
    cout<<"Isi variabel data adalah"<<endl;
    cout<<"-----"<<endl;
    cout<<" Matriks 1 |      Matriks 2 "<<endl;
    for (i=0; i<4; i++)
    {
        for (j=0; j<2; j++)
        {
            cout<<" "<<data[i][j]<<" ";
        }
        cout<<" |      ";
        for (j=0; j<2; j++)
        {
            cout<<data2[i][j]<<" ";
        }
        cout<<endl;
    }
    //proses penjumlahan
    cout<<"\nHasil Penjumlahan Kedua Matriks"<<endl;
    for (i=0; i<4; i++)
    {
        for (j=0; j<2; j++)
        {
            hasil[i][j]=data[i][j]+data2[i][j];
            cout<<"      "<<hasil[i][j]<<" ";
        }
        cout<<endl;
    }
    getch();
}

```

3. Jalankan program tersebut kemudian perhatikan dan analisa outputnya.
4. Modifikasi program prak11.1 dengan nilai input yang diberikan oleh user.
5. Ketikan program berikut ini kemudian simpan file dengan nama prak11-2.cpp.

```

/* File program prak11-2.cpp */
#include <iostream>
#include <conio.h>
using namespace std;

```

```

int main()
{
    cout<<"  Pencarian Data Pada Array 2 Dimensi (MATRIX)\n";
    cout<<"===== \n";
    int a,b;
    cout<<"Masukkan jumlah baris : ";cin>>a;
    cout<<"Masukkan jumlah kolom : ";cin>>b;
    int ar[a][b];

    cout<<"Anda akan menggunakan Matrix "<<a<<"x"<<b<<endl;
    cout<<"----- \n";
    for(int i=0;i<=a-1;i++){
        for(int j=0;j<=b-1;j++){
            cout<<"Masukkan data baris ke-"<<i<<" kolom ke-"<<j<<"
: ";cin>>ar[i][j];
        }
        if(i==a-1){
            cout<<"===== \n";
        }
        else{
            cout<<"----- \n";
        }
    }
    cout<<"Model Matrix\n";
    for(int i=0;i<=a-1;i++){
        for(int j=0;j<=b-1;j++){
            cout<<ar[i][j]<<"\t";
        }
        cout<<endl;
    }
    int cari,tot=0;
    cout<<"Masukkan data yang ingin dicari : ";cin>>cari;
    cout<<"----- \n";
    for(int i=0;i<=a-1;i++){
        for(int j=0;j<=b-1;j++){
            if(ar[i][j]==cari){
                cout<<"Data "<<cari<<" ditemukan pada posisi
["<<i<<"] ["<<j<<"] "<<endl;
                tot+=1;
            }
        }
    }
    cout<<"===== \n";
    if(tot!=0){
        cout<<"Data "<<cari<<" berjumlah "<<tot;
    }
    else{
        cout<<"Data "<<cari<<" tidak ditemukan!!!";
    }
    getch();
}

```

6. Jalankan program tersebut kemudia amati hasil outputnya.
7. Lakukan modifikasi pada program prak11-2 dengan data array bertipe char.
8. Ketikan kode program berikut ini kemudian simpan file dengan nama prak11-3.cpp.

```

/* File program prak11-3.cpp */
#include <iostream>
#include <conio.h>
using namespace std;

int main()
{
    cout<<"  Array Data Mahasiswa";
    cout<<"=====\n";
    int n;
    cout<<"Jumlah Data="; cin>>n;
    string mahasiswa[n][3];
    for (int i=0; i<n; i++){
        cout<<"Data ke-" <<i << endl;
        for (int j=0;j<3;j++){
            if (j==0)
                cout<<"Nim =";
            else if(j==1)
                cout<<"Nama =";
            else
                cout<<"Prodi=";

            cin>>mahasiswa[i][j];
        }

        cout<<"Data mahasiswa yang disimpan" <<endl;
        cout<<"=====" <<endl;
        cout<<"No. Nim      Nama      Prodi" <<endl;

        for (int i=0; i<n;i++){
            cout<<i<<"      "<<mahasiswa[i][0] <<"      "<<mahasiswa[i][1]
<<"      " <<mahasiswa[i][2] <<endl;
        }

        getch();
    }
}

```

9. Jalankan program tersebut kemudian analisa hasil outputnya.

### 11.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Buatlan program perkalian dua buah matrik dengan ketentuan sebagai berikut: 1. Ukuran matrik bebas sesuai dengan input dai user 2. Isi elemen matrik bebesa sesuai dengan input dari user	35 35
2.	CPL-03	CPMK-03	Tuliskan analisa dan penjelasan dari program diatas	30

## 11.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-03	20%		
2.	Praktik	CPL-03	CPMK-03	50%		
3.	Post-Test	CPL-03	CPMK-03	20%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 12:STRUCT

Pertemuan ke : 12

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-03	Mampu mengimplementasi solusi masalah menjadi subrutin, menuliskan subrutin dengan kode bahasa pemrograman, menggunakan ulang (re-use) subrutin pada pemrograman

### 12.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep tentang struct (tipe data struktur) yang meliputi pengertian, deklarasi, pemberian nilai pada elemen struct dan pemanggilannya elemen struct.
2. Menerapkan konsep struct (tipe data srtuktur) kedalam sebuah program komputer dengan C++.

### 12.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-03	Kemampuan mahasiswa dalam menjelaskan dan menerapkan konsep struct meliputi deklarasi, pemberian nilai pada elemen struct, dan pemanggilan elemen struct.
--------	---------	---

### 12.3 TEORI PENDUKUNG

Dlm kebanyakan bahasa Pemrograman memungkinkan kita buat tipe data sendiri. Tipe data buatan / definisi sendiri dalam Bhs C / C++ disebut **Structure**. Structure dapat berisi beberapa tipe data dan Structure lainnya. Konsep dasar dari Structure adalah hampir sama dgn Array kecuali dalam Array semua tipe data sama, maka dalam Structure tipe data dapat berbeda.

Structure adalah turunan dari tipe data yang menggambarkan item data yang berhubungan yang sering disebut komponen / elemen / anggota yang tipe datanya tidak harus sama.

Contoh:

- Tipe Data Mhs

Mhs terdiri : NIM, Nama, Alamat

- Tipe Tgl  
Tgl, Bln, Tahun

Deklarasi Structure didalam bahasa C /C++ secara umum adalah seperti berikut:

```
struct namastructure

    typedata element1;

    typedata element2;

    . . .

};
```

Contoh deklarasi struct:

```
struct Mahasiswa
{
    int    nim;
    char   nama[30];
    char   alamat[50];
};
```

Setelah tipe **Structure** dideklarasikan, kita dpt menggunakan tipe data tsb untuk mendefinisikan suatu variabel. Deklarasi variabel Structure membutuhkan:

- Kunci struct
- Nama tipe **Structure**
- Daftar anggota (nama variabel) dipisahkan dgn koma
- Ditutup dgn semicolon

## 12.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

13 Komputer.

14 Dev C++

## 12.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Jelaskan apa yang dimaksud dengan struct dan pa kegunaanya	20
2.	CPL-03	CPMK-03	Tuliskan kode program untuk deklarasi struct dengan barang dan struktur adalah kode bertipe string, nama bertipe string, harga beli, harga jual, dan stok bertipe integer.	20
			Isi data struct dengan kode: "123" nama: "Buku tulis" Hargabeli = 1500, harga jual: 2500, stok: 12.	30
			Tuliskan kode untuk mencetak elemen struct kode, nama, dan stok.	30

## 12.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-03	Selesaikan langkah praktikum 1 – 9	Hasil praktikum langkah 1 – 9	60
2.	CPL-03	CPMK-03	Selesaikan langkah praktikum 10 – 12	Hasil praktikum langkah 10 – 12	40

### Langkah-Langkah Praktikum:

- Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul.
- Ketik program di bawah ini dan simpan dalam file prak12-1.cpp. Program dibawah ini mendeklarasikan sebuah struct bernama mahasiswa.

```

/* File program prak12-1.cpp */
#include<iostream>
#include<string.h>
#include<conio.h>
using namespace std;

struct mahasiswa
{
    char nama[20];
    int id;
};

void display(mahasiswa s1)
{
    cout<<"Nama : "<<s1.nama<<endl;
    cout<<"ID      : "<<s1.id;
}

int main()
{
    mahasiswa mhs;
    strcpy(mhs.nama, "Ahmad");    //inisialisasi variabel
    mhs.id=123456;
    display(mhs);
    getch();
}

```

- Perhatikan pada variable bertipe struct, elemen-elemen dari struct diakses menggunakan tanda “.” Misalnya, mhs.nama, mhs.id
- Jalankan program di atas dan amati hasilnya.
- Ketik program di bawah ini dan simpan dalam file prak12-2.cpp.

```

/* File program praktikum struct 2 */
#include<iostream>
#include<string.h>
#include<conio.h>

```



```

using namespace std;
struct mahasiswa
{
    char nama[20];
    int id;
};

struct mahasiswa read(); //fungsi prototype

int main()
{
    struct mahasiswa mhs;
    mhs=read();          //memanggil fungsi read
    cout<<"Nama: "<<mhs.nama;
    cout<<"\nID: "<<mhs.id;
    getch();
}

struct mahasiswa read()
{
    struct mahasiswa s1;
    cout<<"Masukkan nama      : ";
    cin>>s1.nama;
    cout<<"Masukkan ID        : ";
    cin>>s1.id;
    cout<<endl;
    return s1;
}

```

6. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya
7. Gabungkan program prak12-1 dan prak12-2 sehingga menjadi program dengan fungsi masing untuk membaca dan menampilkan data dengan input dari user.
8. Ketik program di bawah ini dan simpan dalam file prak11-3.cpp. Program berikut ini menggunakan fungsi dan struct.

```

/* File program praktikum struct 3 */
#include<iostream>
#include<string.h>
#include<conio.h>
using namespace std;

struct nilai
{
    int nim;
    int nilai;
};

nilai masuk ()
{
    nilai n;
    cout<<"Masukkan NIM      : ";
    cin>>n.nim;
    cout<<endl;
    cout<<"Masukkan Nilai      : ";
    cin>>n.nilai;
}

```

```

        return n;
    }

    void tampil(nilai nil)
    {
        cout<<"===== "<<endl;
        cout<<"NIM   : "<<nil.nim<<endl;
        cout<<"Nilai       : "<<nil.nilai<<endl;
    }

    main()
    {
        nilai Nile;
        Nile=masuk();
        tampil(Nile);
        getch;
    }

```

9. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan lakukan analisa pada outputnya.
10. Ketik program di bawah ini dan simpan dalam file prak12-4.cpp. Program di bawah ini menggunakan struct, array dan perulangan sekaligus. Array m berisi data bertipe struct mhs.

```

/* File program prak12-4.cpp */
#include <iostream>
using namespace std;

struct mhs{
    char nama[30];
    int nim, uts, uas, nilai;
};

main()
{
    int n;
    cout<<"Masukkan Jumlah Mahasiswa: ";
    cin>>n;
    mhs m[n];

    for (int i=0; i<n; i++)
    {
        cout<<"Masukkan Nama           : ";
        cin>>m[i].nama;
        cout<<"Masukkan Nim           : ";
        cin>>m[i].nim;
        cout<<"Masukan Nilai UTS : ";
        cin>>m[i].uts;
        cout<<"Masukan Nilai UAS : ";
        cin>>m[i].uas;
        cout<<"\n";
    }
    for (int i=0; i<n; i++)
    {
        m[i].nilai=(m[i].uas+m[i].uts)/2;
        cout<<"Nama           : "<<m[i].nama<<endl;
        cout<<"Nim           : "<<m[i].nim<<endl;
    }
}

```

```

        cout<<"Nilai Akhir : "<<m[i].nilai<<endl<<endl;
    }
    return 0;
}

```

11. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.
12. Ketik program di bawah ini dan simpan dalam file prak11-5.cpp. Jalankan dan amati hasil outputnya!

```

/* File program prak12-5 */
#include <iostream>
using namespace std;
int main()
{
    struct biodata
    {
        // tipe data dan variabel dikumpulkan disini
        char nama[30], hobi[40], asal[20], sekolah[30];
        int umur, status;
    };
    biodata bio;
    // Input menggunakan struct
    cout << "Masukan nama anda : ";
    cin.getline(bio.nama, 30);

    cout << "Masukan hobi anda : ";
    cin.getline(bio.hobi, 40);

    cout << "Masukan umur anda : ";
    cin >> bio.umur;

    cout << "Masukan kota asal anda : ";
    cin >> bio.asal;

    cout << "Masukan sekolah anda : ";
    cin >> bio.sekolah;

    cout << "Masukan status anda (1 apabila single, 2 apabila
double): ";
    cin >> bio.status;

    // output menggunakan struct
    cout<<"\nData Kamu"<<endl;
    cout<<"======"<<endl;
    cout<<"Nama saya "<<bio.nama<<" " <<endl;
    cout<<"Hobi saya adalah "<<bio.hobi<<" " <<endl;
    cout<<"Saya berusia "<<bio.umur<<" tahun. " <<endl;
    cout<<"Saya berasal dari kota "<<bio.asal<<" " <<endl;
    if (bio.status==1)
    {
        cout<<"Saat ini saya masih single :(";
    }
    else
    {
        cout<<"Maaf, sudah ada yang punya :)";
    }
}

```

```

    }
    return 0;
}

```

## 12.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Buatlah sebuah program sederhana untuk mendata dan menampilkan informasi buku dengan menggunakan struct. Informasi buku yang ditampilkan adalah: kode buku, judul buku, penulis, penerbit, dan harga.	70
2.	CPL-03	CPMK-03	Tuliskan analisa dan penjelasan dari program diatas	30

## 12.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-03	20%		
2.	Praktik	CPL-03	CPMK-03	50%		
3.	Post-Test	CPL-03	CPMK-03	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 13: POINTER

**Pertemuan ke** : 13

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-03	Mampu mengimplementasi solusi masalah menjadi subrutin, menuliskan subrutin dengan kode bahasa pemrograman, menggunakan ulang (re-use) subrutin pada pemrograman

### 13.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep pointer dan pembuatannya meliputi deklarasi, mengisi data pointer, dan mengakses pointer dan alokasi memorinya.
2. Menerapkan konsep pointer ke dalam sebuah program komputer dengan C++

### 13.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-01	CPMK-03	Kemampuan mahasiswa dalam menjelaskan dan menerapkan konsep pointer dalam sebuah program komputer dengan C++.
--------	---------	---

### 13.3 TEORI PENDUKUNG

Pointer adalah variable yang menunjuk ke alamat memori dari variable lain. Karena pointer menunjuk alamat dari sebuah variable, maka pointer akan berisi alamat memori dan bukan berisi nilai. Setiap variable atau fungsi pada kode program menunjuk alamat tertentu pada memori. Pointer bertugas untuk menunjuk alamat-alamat pada memori tersebut.

Jika kita mempunyai sebuah variable dengan tipe data tertentu, maka untuk mendapatkan alamat dari variable tersebut adalah dengan menggunakan operator dereference dengan menggunakan tanda **&**. Alamat inilah yang kemudian akan disimpan ke dalam variable yang bertipe pointer. Untuk

mendeklarasikan variable sebagai pointer, kita hanya menambahkan operator *reference* atau tanda \* di depan nama variable. Deklarasi Pointer didalam bahasa C /C++ secara umum adalah seperti berikut:

**TipeData \*NamaPointer;**

Contoh deklarasi pointer dalam program:

```
int *p;      → mendeklarasikan pointer p dengan tipe data integer
long *x;     → mendeklarasikan pointer x dengan tipe data long
...
dll
```

Jika kita mempunyai sebuah variable yang bertipe int (misal x), maka kita dapat memerintahkan sebuah pointer untuk menunjuk ke alamat yang ditempati oleh variable x. Untuk melakukan ini, kita dapat menggunakan kode berikut:

```
int a = 10; → mendeklarasikan variable a dengan nilai 10
int *p;     → mendeklarasikan pointer p dengan tipe data integer
*p = &a;    → memerintahkan pointer p untuk menunjuk ke alamat dari
             variable a
```

Gambar berikut menjelaskan mengenai contoh kode program di atas. Pada kode program di atas dilakukan deklarasi variable a yang bertipe int dengan nilai 10. Kemudian dilakukan deklarasi pointer p (\*p) yang bertipe int. Kemudian pointer p menunjuk ke alamat variable a (&a). Pada saat pointer p menunjuk ke alamat variable a, maka pointer p akan memiliki nilai yang sama dengan variable a dan p akan sama dengan &a. p dan &a sama-sama berisi alamat memori. Kemudian variabel b menunjuk ke pointer p yang memiliki nilai dari variable a (10).

Nama Variabel	a	p
Nilai Variabel	10	0x6ffe0c
Alamat Memori	0x6ffe0c	0x6ffe00

Gambar 13.1 Ilustrasi Pointer

Dengan menggunakan cara di atas, kita dapat mengakses dan memanipulasi nilai variable b melalui pointer p. Proses mengakses nilai a melalui pointer p ini yang kita sebut dengan dereference pointer.

## 13.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

- 14 Komputer.
- 15 Dev C++.

## 13.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Apa yang dimaksud dengan pointer dan apa kelebihan pointer di banding dengan tipe data lainnya.	30
2.	CPL-03	CPMK-03	Tuliskan kode program untuk mendeklarasikan pointer bertipe integer, memberikan nilai 5 pada pointer, mencetak isi pointer, dan mencetak alamat pointer.	70

## 13.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-03	Selesaikan langkah praktikum 1 – 7	Hasil praktikum langkah 1 – 7	60
2.	CPL-03	CPMK-03	Selesaikan langkah praktikum 8 – 11	Hasil praktikum langkah 8 – 11	40

### Langkah-Langkah Praktikum:

- Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul
- Ketik kode program di bawah ini dan simpan ke dalam file prak12-1.cpp. Perhatikan bahwa tanda & digunakan untuk mengambil alamat memori dari suatu variabel.

```
#include <iostream>
using namespace std;

int main()
{
    int a = 5;
    cout<<"Alamat Variabel a adalah : "<<&a<<endl;
    cout<<"Nilai Variabel a adalah : "<<a<<endl;
    return 0;
}
```

- Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.
- Ketik kode program di bawah ini dan simpan dalam file prak12-2.cpp. Pada program ini, variable pointer yang ditandai dengan \* akan diisi alamat memory dari variabel.

```
#include<iostream>
using namespace std;

int main()
{
    int *p;
    int x;

    p = &x;
    x = 10;
    cout<<"Nilai x adalah "<<x<<endl;
    cout<<"Nilai *p adalah "<<*p<<endl;
    cout<<"Nilai p adalah "<<p<<endl;
    cout<<"Nilai &x adalah "<<&x<<endl<<endl;
    *p = 200;
    cout<<"Nilai x adalah "<<x<<endl;
    cout<<"Nilai *p adalah "<<*p<<endl;
    cout<<"Nilai p adalah "<<p<<endl;
    cout<<"Nilai &x adalah "<<&x<<endl;
    return 0;
}
```



5. Jalankan program tersebut secara berulang dengan nilai variable yang dirubah atau ditambah. Perhatikan perubahan pada outputnya.
6. Ketik kode di bawah ini dan simpan dalam file prak12-3.cpp. Berikut ini contoh program dengan pointer dan alokasi memori tersendiri menggunakan new .

```
#include<iostream>
using namespace std;

int main()
{
    int x, *y;
    y = new int;
    x = 15;
    *y = 20;
    cout<<"Nilai x "<<x<<endl;
    cout<<"Alamat memori x = "<<&x<<endl;
    cout<<"Nilai y = "<<*y<<endl;
    cout<<"Alamat memori y = "<<y<<endl;
    delete y;
    cout<<"Setelah di-delete, nilai y adalah "<<*y<<endl;
    cout<<"Alamat memori y adalah "<<y<<endl;
    return 0;
}
```

7. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.
8. Ketik kode program di bawah ini, dan simpan dalam file prak12-4.cpp. Pada program ini dibuat sebuah pointer yang bertipe void. Pointer void adalah pointer yang belum jelas akan menyimpan alamat dari variable bertipe apa, oleh karena itu pointer ini dapat

```
#include <iostream>
using namespace std;

int main()
{
    int i = 10;
    float f = 0.25;
    double d = 123.123;

    void *p =NULL;                //deklarasi void pointer

    p = &i;                        //menunjuk ke alamat int i
    cout<<"p menunjuk ke alamat i"<<endl;
    cout<<"Nilai *p: "<<*((int*)p)<<endl;
    cout<<"Nilai p: "<<p<<endl<<endl;
    p = &f;                        //menunjuk ke alamat float f
    cout<<"p menunjuk ke alamat f"<<endl;
    cout<<"Nilai *p: "<<*((float*)p)<<endl;
    cout<<"Nilai p: "<<p<<endl<<endl;
    p = &d;                        //menunjuk ke alamat double d
    cout<<"p menunjuk ke alamat d"<<endl;
    cout<<"Nilai *p: "<<*((double*)p)<<endl;
}
```

```
cout<<"Nilai p: "<<p<<endl<<endl;
    return 0;
}
```

9. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.
10. Ketik kode program berikut ini, dan simpan dalam file prak12-5.cpp. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.

```
#include <iostream>
using namespace std;

int main()
{
    int nilai1=5, nilai2=15;
    int *p1, *p2;
    p1 = &nilai1;    //p1 = alamat nilai1

    p2 = &nilai2;    //p2 = alamat nilai2

    cout<<"Nilai awal"<<endl;
    cout<<"Nilai 1 adalah "<<nilai1<<endl;
    cout<<"Nilai 2 adalah "<<nilai2<<endl;

    *p1 = 10; //nilai alamat yang ditunjuk p1 = 10
    cout<<"\nSetelah *p1 = 10"<<endl;
    cout<<"Nilai 1 adalah "<<nilai1<<endl;
    cout<<"Nilai 2 adalah "<<nilai2<<endl;

    *p2 = *p1; /*nilai alamat yang ditunjuk oleh p2 =
                nilai alamat yang ditunjuk oleh p1*/
    cout<<"\nSetelah *p2 = *p1"<<endl;
    cout<<"Nilai 1 adalah "<<nilai1<<endl;
    cout<<"Nilai 2 adalah "<<nilai2<<endl;

    p1 = p2; //p1=p2 (mengkopi nilai pointer)
    cout<<"\nSetelah p1 = p2"<<endl;
    cout<<"Nilai 1 adalah "<<nilai1<<endl;
    cout<<"Nilai 2 adalah "<<nilai2<<endl;

    *p1 = 20; // nilai alamat yang ditunjuk p1 = 20
    cout<<"\nSetelah *p1 = 20"<<endl;
    cout<<"Nilai 1 adalah "<<nilai1<<endl;
    cout<<"Nilai 2 adalah "<<nilai2<<endl;

    return 0;
}
```

11. Jalankan program tersebut secara berulang dengan nilai variable yang diubah-ubah. Perhatikan dan analisislah korelasi input dan outputnya.

### 13.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	Buatlah sebuah program pointer sederhana yang menggunakan operator reference (*) dan operator dereference (&).	60
2.	CPL-03	CPMK-03	Buatlah analisa dan penjelasan dari program diatas	40

### 13.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-03	20%		
2.	Praktik	CPL-03	CPMK-03	50%		
3.	Post-Test	CPL-03	CPMK-03	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## PRAKTIKUM 14: AKSES FILE

**Pertemuan ke** : 14

**Total Alokasi Waktu** : 180 menit

- Materi : 15 menit
- Pre-Test : 20 menit
- Praktikum : 115 menit
- Post-Test : 30 menit

**Total Bobot Penilaian** : 100%

- Pre-Test : 20 %
- Praktik : 50 %
- Post-Test : 30 %

**Pemenuhan CPL dan CPMK:**

CPL-03	Mampu menerapkan konsep teoritis bidang area Informatika terkait matematika dasar dan ilmu komputer untuk memodelkan masalah dan meningkatkan produktivitas
CPMK-03	Mampu mengimplementasi solusi masalah menjadi subrutin, menuliskan subrutin dengan kode bahasa pemrograman, menggunakan ulang (re-use) subrutin pada pemrograman

### 14.1 DESKRIPSI CAPAIAN PEMBELAJARAN

Setelah mengikuti praktikum ini mahasiswa diharapkan mampu:

1. Menjelaskan konsep penggunaan file dengan operasi baca file dan menulis ke file
2. Menerapkan konsep penggunaan file dengan operasi penulisan dan pembacaan pada program komputer dengan menggunakan C++.

### 14.2 INDIKATOR KETERCAPAIAN PEMBELAJARAN

Indikator ketercapaian diukur dengan:

CPL-03	CPMK-03	Kemampuan mahasiswa dalam menjelaskan dan menerapkan akses file (penulisan dan pembacaan) ke dalam sebuah program dengan menggunakan C++
--------	---------	--

### 14.3 TEORI PENDUKUNG

Dalam hampir semua bahasa pemrograman termasuk C/C++ terdapat fasilitas untuk mengakses file external (yang tersimpan dalam media simpan external /disk). Dalam bahasa C/C++ referensi yang digunakan untuk mengakses file adalah dengan library standard `<stdio.h>`, yang mana library tersebut menyediakan beberapa fungsi manipulasi file. Untuk mengakses file digunakan deklarasi umum file seperti berikut :

```
FILE * internal_file_name;
```

- Menyatakan bahwa `internal_file_name` menunjuk pada struktur file
- Jika ada lebih dari 1 file, maka setiap file membutuhkan FILE pointer sendiri

- `internal_file_name` merupakan nama file eksternal yg digunakan untuk menyimpan dalam media simpan eksternal
- jika file tidak berada pada direktori aktif maka harus disebutkan alamat folder/direktornya

Sebelum kita memproses file, kita harus membuka atau membuat file Syntax untuk fungsi

`fopen()`:

```
internal_file_name = fopen(external_file_name, OpenMode);
```

Fungsi `fopen()` membutuhkan 2 argumen : nama file dan mode file untuk daftar mode buka file yg dapat dihubungkan dgn `fopen()`.

- “r” : untuk membuka file pd mode input shg data dpt dibaca dari file tersebut. File yg dibuka harus berada dalam disk yg aktif
- “w” : berarti file disediakan untuk output dan untuk ditulisi data
- “a” : berarti file untuk ditambahi data pada akhir file
- “r+” : file dibuka untuk diupdate, dalam arti bahwa file akan dibaca dan tulis/tambahi data tanpamerusak isi
- “w+” : menimpa file yg sudah ada atau membuka file baru untuk diupdate
- “a+” : membuka file untuk update Untuk contoh akan disampaikan dlm langkah praktikum.

Selain fungsi `fopen()` yang sering dibutuhkan adalah `fgetc()` . Fungsi `fgetc()` merupakan fungsi yg akan membaca character dari file

Syntax :

```
variable = fgetc(internal_file_name);
```

Dimana :

- Variabel merupakan nama variabel yg akan digunakan untuk menampung hasil kembalian dari `fgetc()` .
- `Internal_file_name` merupakan pointer untuk membuka file

Fungsi lainnya adalah `fputc()` yang merupakan fungsi yang akan menuliskan nilai character ke dalam file

Syntax:

- Menggunakan variabel character ch:

```
fputc(ch, internal_file_name);
```

- Menggunakan nilai konstanta character :

```
fputc('a', internal_file_name);
```

## 14.4 HARDWARE DAN SOFTWARE

Hardware dan software yang digunakan dalam praktikum ini yaitu:

- 1 Komputer.
- 2 Dev C++.

## 14.5 PRE-TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	jelaskan bagaimana mendeklarasikan sebuah variabel yang digunakan untuk mengakses file eksternal program.	40

2.	CPL-03	CPMK-03	Jelaskan bagaimana langkah untuk membaca sebuah file dan fungsi apa yang digunakan.	30
3	CPL-03	CPMK-03	Jelaskan bagaimana langkah menuliskan data kedalam sebuah file dan fungsi apa yang digunakan.	30

## 14.6 LANGKAH PRAKTIKUM

**Aturan Penilaian (Total Skor: 100):**

No	CPL	CPMK	Pertanyaan	Dokumen Pendukung	Skor
1.	CPL-03	CPMK-03	Selesaikan langkah praktikum 1 – 4	Hasil praktikum langkah 1 – 4	40
2.	CPL-03	CPMK-03	Selesaikan langkah praktikum 5 – 10	Hasil praktikum langkah 5 – 10	60

### Langkah-Langkah Praktikum:

- Ikuti langkah-langkah dibawah ini, ketik sendiri dan jangan COPY-PASTE dari modul
- Buatlah file bertipe .txt berisi teks dibawah ini, simpan dengan nama file “dokumen.txt” di folder yang sama dengan file program C/C++ Anda

Saya mahasiswa semester 1 program studi Teknik Informatika Fakultas Teknologi Industri Universitas Ahmad Dahlan.

- Ketik kode program di bawah ini, simpan ke dalam file prak14-1.c.

```
#include <stdio.h>

#include <stdlib.h>
int main()
{
    FILE *filePointer;
    char ch;
    filePointer = fopen("dokumen.txt", "r"); // akses
    read
    if (filePointer == NULL)
    {
        printf("File is not available \n");
    }
    else
    {
        while ((ch = fgetc(filePointer)) != EOF)
        {
            printf("%c", ch);
        }
    }
}
```

```

        fclose(filePointer);
        return 0;
    }

```

4. Jalankan program di atas. Perhatikan outputnya. Perhatikan kode di atas.

Baris `filePointer = fopen("dokumen.txt", "r");` berfungsi untuk membuka file dengan mode "read" atau membaca.

Baris `fgetc(filePointer)` berfungsi untuk membaca setiap karakter pada file dokumen.txt.

Baris `while ((ch = fgetc(filePointer)) != EOF)` maksudnya adalah baca setiap karakter pada file dokumen.txt selama karakter yang dibaca bukan EOF (End of File)

Baris `fclose(filePointer)` digunakan untuk menutup file yang telah selesai dibaca.

5. Buatlah sebuah file "alphabet.txt" dan biarkan kosong. Ketik kode program di bawah ini, simpan ke dalam file prak14-2.c.

```

/* fputc example: alphabet writer */

#include <stdio.h>

int main ()
{
    FILE * pFile;
    char c;

    pFile = fopen ("alphabet.txt","w"); // akses write
    if (pFile!=NULL) {

        for (c = 'A' ; c <= 'Z' ; c++)
            fputc ( c , pFile );

        fclose (pFile);
    }
    return 0;
}

```

6. Perhatikan kode di atas lihatlah hasil outputnya, `fputc` berfungsi untuk menulis sebuah karakter ke dalam file yang telah dibuka dengan mode "write" (menulis).
7. Buatlah sebuah file "mylog.txt" dan biarkan kosong. Ketik kode program di bawah ini, kemudian simpan ke dalam file prak14-3.c.



```

/* fputs example */
#include <stdio.h>

int main ()
{
    FILE * pFile;
    char sentence [256];

    printf ("Enter sentence to append: ");
    fgets (sentence,256,stdin);
    pFile = fopen ("mylog.txt","a"); // akses append
    fputs (sentence,pFile);
    fclose (pFile);
    return 0;
}

```

8. Jalankan program di atas, perhatikan outputnya. Perhatikan kode berikut ini.

```
fgets (sentence,256,stdin);
```

Baris kode tersebut digunakan untuk mengambil input dari user dalam bentuk string. String yang diinputkan oleh user kemudian digabungkan dengan isi file mylog.txt dengan perintah berikut.

```
fputs (sentence,pFile);
```

9. Tulis kode program di bawah ini dan simpan dalam file `prak14-4.c`. Perhatikan kode program di bawah ini, program akan menerima input dari user kemudian menulis input tersebut ke dalam file `myfile.txt`

```

/* fprintf example */
#include <stdio.h>

int main ()
{
    FILE * pFile;
    int n;
    char name [100];

    pFile = fopen ("myfile.txt","w");
    for (n=0 ; n<3 ; n++)
    {
        puts ("please, enter a name: ");
        gets (name);
        fprintf (pFile, "Name %d [%-10.10s]\n",n+1,name);
    }
    fclose (pFile);

    return 0;
}

```

10. Jalankan program di atas, perhatikan outputnya. Apa yang tertulis di dalam file `myfile.txt`?

## 14.7 POST TEST

Jawablah pertanyaan berikut (**Total Skor: 100**):

No	CPL	CPMK	Pertanyaan	Skor
1.	CPL-03	CPMK-03	<p>Buatlah sebuah program sederhana yang menyerupai sebuah mesin kasir.</p> <p>Menu utama pada program tersebut adalah 1) Masukkan barang belanja 2) Keluar.</p> <p>Selama user memilih menu no 1 maka program akan meminta input Nama Barang, Kuantitas (jumlah barang), Harga satuan. Setelah itu program akan menghitung Total biaya belanja untuk satu item barang yang diinputkan.</p> <p>Jika user memilih menu no 2 maka program akan menghitung keseluruhan total belanja barang dan program akan menge-print struk belanja kemudian setelah itu keluar dari program.</p> <p>Implementasikan program tersebut, dan cetaklah struk belanja pada File "Struk.txt"!</p>	<p>40</p> <p>60</p>

## 14.8 HASIL CAPAIAN PRAKTIKUM

Diisi oleh asisten setelah semua assessment dinilai.

No	Bentuk Assessment	CPL	CPMK	Bobot	Skor (0-100)	Nilai Akhir (Bobot x Skor)
1.	Pre-Test	CPL-03	CPMK-01	20%		
2.	Praktik	CPL-03	CPMK-01	50%		
3.	Post-Test	CPL-03	CPMK-01	30%		
<b>Total Nilai</b>						

**LEMBAR JAWABAN PRE-TEST DAN POST-TEST PRAKTIKUM**

<b>Nama :</b> <b>NIM :</b>	<b>Asisten:</b> <b>Paraf Asisten:</b>	<b>Tanggal:</b> <b>Nilai:</b>
-------------------------------	--	----------------------------------

--

## DAFTAR PUSTAKA

1. C++ How to Program, 8/E; Paul Deitel, Harvey Deitel; Prenticed Hall, 2012
2. Learn to Program with C++, John Smiley, McGraw Hill, 2006
3. Theory and Program of Programming with C++; JOHN R. HUBBARD; McGraw Hill
4. [www.cplusplus.com](http://www.cplusplus.com)
5. [www.tutorialspoint.com](http://www.tutorialspoint.com)
6. [www.cppreference.com](http://www.cppreference.com)

