



# PEMROGRAMAN DASAR

## #01 – Pengantar

Oleh: Fahmi Ruziq, S.T., M.Kom.



# Perkenalan Diri Dosen

# Perkenalan Diri Dosen

- Nama lengkap : Fahmi Ruziq, S.T., M.Kom.
- Latar Belakang Pendidikan:
  - **2014** - S1 Teknik Informatika, Universitas Serambi Mekkah, Banda Aceh.
  - **2020** - S2 Teknik Informatika, Universitas Sumatera Utara, Medan.
  - **(on going)** S3 Ilmu Komputer, Universitas Sumatera Utara, Medan.

# Perkenalan Diri Dosen

- **Pengalaman Kerja:**
  - 2012 – 2015 : Customer Service di Bank BRI Cabang Banda Aceh.
  - 2016 – 2021: Supervisor Warehouse di PT. Mitra Timur Lestari, Medan.
  - 2021 – 2022: Programmer/Software Developer CV. Sumatera Indah Production, Medan.
  - 2020 – sekarang : Dosen Tetap Universitas Battuta, Medan.

# Perkenalan Diri Dosen

- **Jabatan di Universitas Battuta:**
  - 2021 – sekarang : Koordinator UKM Programming.
  - 2022 – 2025: Ketua Program Studi Sistem Informasi.
- **No. Handphone:** +62 852 7574 1342
- **Email:** [fahmiruziq89@gmail.com](mailto:fahmiruziq89@gmail.com)



DIKTISAINTEK  
BERDAMPAK



# Gambaran Umum Mata Kuliah

# Gambaran Umum Mata Kuliah

- **Deskripsi Singkat Mata Kuliah:**

Mata Kuliah Pemrograman Dasar membahas tentang konsep dasar pemrograman dan penggunaan bahasa C++ untuk mengembangkan sistem informasi. Mata kuliah ini membantu mahasiswa memahami variabel, tipe data, operator, kontrol struktur, fungsi, array, pointer, struktur data dasar, dan *object-oriented programming*, serta mengembangkan program C++ yang terstruktur dan efektif.

# **Gambaran Umum Mata Kuliah**

- Capaian Pembelajaran Lulusan (CPL):**

Mampu mengembangkan sistem informasi yang mencakup pemrograman, analisis sistem, dan penggunaan teknologi terkini dengan menggunakan metode dan tools yang sesuai.

# Gambaran Umum Mata Kuliah

- **Capaian Pembelajaran Mata Kuliah (CPMK):**

- a. Mampu memahami konsep dasar pemrograman, termasuk variabel, tipe data, operator, dan kontrol struktur.
- b. Mampu menguasai sintaks dasar bahasa C++, termasuk deklarasi variabel, penulisan fungsi, dan penggunaan kontrol struktur.
- c. Mampu menggunakan tipe data dasar di C++, termasuk integer, float, dan karakter.

# Gambaran Umum Mata Kuliah

- **Capaian Pembelajaran Mata Kuliah (CPMK):**

- d. Mampu mengimplementasikan kontrol struktur di C++, termasuk if-else, switch, dan loop.
- e. Mampu menggunakan fungsi di C++, termasuk deklarasi fungsi, penggunaan parameter, dan pengembalian nilai.
- f. Mampu menguasai konsep array dan pointer di C++, termasuk deklarasi array, penggunaan pointer, dan manipulasi data.

# Gambaran Umum Mata Kuliah

- **Capaian Pembelajaran Mata Kuliah (CPMK):**
  - g. Mampu menggunakan struktur data dasar di C++, termasuk struct dan class.
  - h. Mampu menguasai konsep object-oriented programming di C++, termasuk penggunaan class, objek, dan inheritance.
  - i. Mampu mengembangkan program C++ yang terstruktur dan efektif, termasuk penggunaan konsep-konsep yang telah dipelajari dan pengembangan program yang baik.

# Gambaran Umum Mata Kuliah

- **Ringkasan CPMK:**

- Pertemuan 1-2 : Konsep Dasar Pemrograman
- Pertemuan 3 : Sintaks Dasar Bahasa C++
- Pertemuan 4 : Tipe Data Dasar di C++
- Pertemuan 5-6 : Kontrol Struktur di C++
- Pertemuan 7 : Fungsi di C++
- Pertemuan 8 : Ujian Tengah Semester

# Gambaran Umum Mata Kuliah

- **Ringkasan CPMK:**

- Pertemuan 9-10 : Array dan Pointer di C++
- Pertemuan 11 : Struktur Data Dasar di C++
- Pertemuan 12-13 : Object-Oriented Programming di C++
- Pertemuan 14-15 : Pengembangan Program C++ yang Terstruktur dan Efektif
- Pertemuan 16 : Ujian Akhir Semester

# Gambaran Umum Mata Kuliah

- **Penilaian (Khusus MK ini):**

- Kehadiran/Quiz : 10%
- Praktek/Tugas : 15%
- Unjuk Kerja/Presentasi : 20%
- UTS : 20%
- UAS : 20%
- Tes Lisan (Tugas Kelompok) : 15%
- **Total : 100%**

# Tools yang Digunakan

- IDE (Integrated Development Environment)  
→ DEV C++



# Konsep Dasar Pemrograman

# Pustaka

- Bjarne Stroustrup – Programming: Principles and Practice Using C++ (2024)
- Deitel & Deitel – C: How to Program (2023)
- Tony Gaddis – Starting Out with Programming Logic and Design (2019)

# Masalah

- Hidup manusia tidak lepas dari masalah.
- Bahkan, hidup itu sendiri dapat dipandang sebagai rangkaian proses dalam menyelesaikan berbagai permasalahan.
- Dalam konteks ilmu komputer, masalah diartikan sebagai pertanyaan atau tugas yang memerlukan jawaban atau solusi.

# Masalah

- Masalah atau persoalan (problem) adalah pertanyaan atau tugas yang kita cari jawabannya.
- Contoh beberapa masalah sehari-hari:
  - mengurutkan kartu antrian pasien di rumah sakit,
  - mengelola daftar buku bagi pustakawan,
  - mencari nilai tertinggi mahasiswa bagi seorang dosen, dll.

# Algoritma

- Untuk masalah dengan instansiasi yang kecil, kita dapat menemukan solusinya dengan mudah dan cepat.
- Bagaimana kalau instansiasi masalah berukuran besar?
- Misalnya pada masalah pengurutan jika  $n = 1000$  jelas tidak mudah mengurutkan data sebanyak itu.

# Algoritma

- Oleh karena itu kita perlu menuliskan prosedur yang berisi langkah-langkah pengurutan sehingga prosedur dapat ‘dijalankan’ oleh sebuah pemroses (komputer, manusia, robot, dsb) untuk menghasilkan solusi setiap instansiasi masalah pengurutan.
- Prosedur yang berisi langkah-langkah penyelesaian masalah disebut dengan **algoritma**.

# Algoritma

- Jadi, algoritma adalah urutan langkah-langkah untuk memecahkan suatu masalah.
- Definisi lain:

Algoritma adalah deretan instruksi yang jelas untuk memecahkan masalah, yaitu untuk memperoleh keluaran yang diinginkan dari suatu masukan dalam jumlah waktu yang terbatas.

# Algoritma

- Langkah-langkah mengurutkan 10 buah kartu sbb:
  - Cari kartu dengan nomor terkecil diantara kartu yang tersisa.
  - Tempatkan kartu tersebut pada posisi yang tepat.
  - Ulangi kembali dari langkah 1 sebanyak (10-1) kali.

# Algoritma

- Langkah-langkah membuat kopi instant:
  - Buka kemasan kopi sachet yang sudah dibeli.
  - Setelah itu tuangkan kopi ke dalam gelas.
  - Tuangkan air panas ke dalam gelas.
  - Lalu aduk hingga merata.
  - Kopi instant siap dinikmati.

# Algoritma

- Di dalam literatur klasik terdapat persoalan yang dinamakan **Water Jug Problem**.
- Misalkan Anda mempunyai 2 buah ember, masing-masing bervolume 5 liter dan 3 liter. Anda diminta mendapatkan air dari sebuah sungai sebanyak 4 liter dengan menggunakan bantuan kedua ember tersebut (tanpa peralatan lain).

# Algoritma

- Anda boleh memindahkan air dari satu ember ke ember lain, membuang seluruh air dari ember, dan lain sebagainya. Bagaimana algoritmanya ?



5 liter



3 liter

# Algoritma

- Penyelesaian masalah:
  - Isi penuh ember 3 liter dengan air.
  - Tuangkan air dari ember 3 liter ke dalam ember 5 liter.
  - Isi penuh lagi ember 3 liter dengan air.
  - Tuangkan air dari ember 3 liter ke dalam ember 5 liter hingga penuh.
  - Buang seluruh air dari ember 5 liter.
  - Tuangkan sisa air dari ember 3 liter ke dalam ember 5 liter.
  - Isi penuh lagi ember 3 liter dengan air.
  - Tuangkan semua air dari ember 3 liter ke dalam ember 5 liter.

# Program dan Pemrograman

- Algoritma baru efektif jika dijalankan oleh sebuah pemroses (processor).
- Pemroses membaca setiap instruksi di dalam algoritma lalu mengerjakannya.
- Suatu pemroses harus:
  - Mengerti setiap langkah dalam algoritma.
  - Mengerjakan operasi yang bersesuaian dengan langkah tersebut.

# Program dan Pemrograman

- Kita memfokuskan pemroses algoritma adalah komputer.
- Komputer adalah alat bantu untuk menjalankan perintah-perintah di dalam algoritma yang telah ‘dimasukkan’ ke dalamnya.
- Agar komputer mengerti perintah yang dimaksudkan, maka perintah tersebut harus ditulis dalam bahasa yang dipahami olehnya.

# Program dan Pemrograman

- Oleh karena itu, algoritma harus ditulis dalam bahasa khusus, yaitu **bahasa komputer**.
- Algoritma yang ditulis dalam bahasa komputer dinamakan **program**.
- Bahasa komputer yang digunakan dalam menulis program dinamakan **bahasa pemrograman** (*programming language*).

# Program dan Pemrograman

- Orang yang membuat program komputer disebut **pemrogram (programmer)**.
- Dan kegiatan merancang dan menulis program disebut **pemrograman (programming)**.
- Di dalam pemrograman ada aktivitas menulis kode program, kegiatan ini dinamakan **coding**.

# Program dan Pemrograman

- Belajar memprogram **tidak sama** dengan belajar bahasa pemrograman.
- Belajar memprogram berarti mempelajari **metodologi pemecahan masalah**, kemudian menuliskan algoritma pemecahan masalah dalam **notasi tertentu** .

# Program dan Pemrograman

- Sedangkan belajar bahasa pemrograman berarti belajar memakai satu bahasa komputer, aturan tata bahasa, instruksi-instruksinya, tata cara pengoperasian *compiler*-nya, dan memanfaatkan instruksi-instruksi tersebut untuk membuat program yang ditulis hanya dalam bahasa itu saja.

# Program dan Pemrograman

- Bahasa pemrograman secara umum:

- Low Level Language

Bahasa jenis ini dirancang agar setiap instruksinya langsung dikerjakan oleh komputer tanpa harus melalui penerjemah (*translator*). Contohnya adalah bahasa mesin (*machine language*) dan *assembly*. Bahasa mesin adalah kumpulan kode biner (0 dan 1).

# Program dan Pemrograman

- **High Level Language**

Bahasa jenis ini membuat program menjadi lebih mudah dipahami karena lebih dekat ke bahasa manusia (Inggris). Kelemahannya, program harus diterjemahkan terlebih dahulu oleh translator (kompilator atau *compiler*) ke dalam bahasa mesin agar dapat dipahami dan dikerjakan oleh komputer.

- **Contohnya** adalah C, C++, C#, Pascal, Basic, Fortran, Java, PHP, Visual Basic .Net, Oracle, Phyton, dll.



**Terima kasih.**