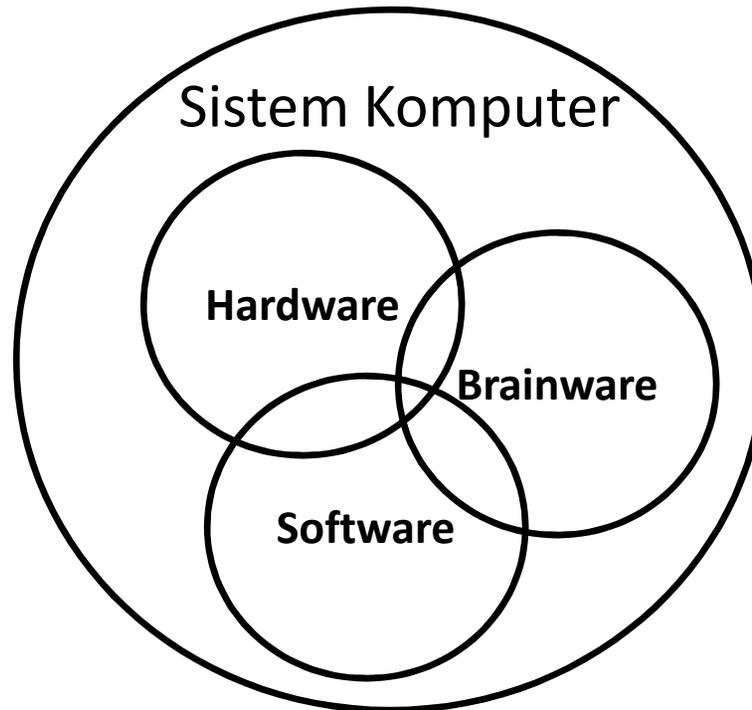


Pertemuan 1

**Konsep Dasar Pemrograman
Berorientasi Objek / PBO**

**(Object Oriented
Programming / OOP)**

Pengertian Pemrograman Berorientasi Objek



PEMROGRAMAN (PROGRAMMING)

Kegiatan menulis kode-kode program (coding) dan mengkompilasinya agar dapat dimengerti oleh komputer dengan menggunakan suatu bahasa pemrograman (mis : Java, C ++, Pascal dll)

BERORIENTASI

**Orientasi (e: Orientation) dalam kamus artinya
“mengarahkan” atau “pendekatan”**

OBJEK

**sesuatu yang mempunyai identitas yang uniq dan
nilai atribut tertentu, atau sesuatu yang bisa kita
cirikan padanya suatu sifat tertentu. (ciri khas)**

PEMROGRAMAN BERORIENTASI OBJEK

merupakan paradigma pemrograman yang berorientasikan kepada objek. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek.

Bandungkan dengan logika pemrograman terstruktur. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya,

Model data berorientasi objek dikatakan dapat memberi fleksibilitas yang lebih, kemudahan mengubah program, dan digunakan luas dalam teknik peranti lunak skala besar. Lebih jauh lagi, pendukung OOP mengklaim bahwa OOP lebih mudah dipelajari bagi pemula dibanding dengan pendekatan sebelumnya, dan pendekatan OOP lebih mudah dikembangkan dan dirawat.

Kenapa OOP?

- Dengan OOP, kode-kode yang kita buat menjadi lebih rapih dan terstruktur.
- Dengan OOP, proses reuse kode-kode yang kita buat untuk project yang hampir sama, mudah, karena kode yang kita buat rapih dan terstruktur. Sehingga untuk mengubah atau reuse menjadi lebih mudah.
- Konsepnya per modul. Jika error pasti mudah diketahui karena terbaca dari fungsi yang kita panggil. Itulah maksudnya konsep per modul. Sehingga dengan OOP, kita dimudahkan untuk membuat dan membaca kode kita (efisiensi kode).
- Konsep OOP juga memudahkan kita untuk menganalisa program yang kita akan buat. Ini akan sangat terasa kalau kita membuat program besar dan rumit

Beberapa jenis teknik pemrograman yang telah pernah ada :

- **unstructured programming**
- ***procedural programming***
- ***Modular Programing***
- ***object-oriented programming***

Unstructured programming

Teknik memrogram dengan cara menuliskan program hanya pada satu program utama. Di dalam program utama tersebut, keseluruhan baris programnya berisi urutan perintah atau statemen untuk memodifikasi dan mengolah data.



procedural programming

- Serangkaian tugas diselesaikan dalam bentuk fungsi atau prosedur.
- Cara pandang → program adalah suatu urutan instruksi.
- Programmer harus me-break down suatu problem menjadi sub problem yang lebih simple.
- Fungsi dan prosedur menjadi fokus utama.
- Fungsi dan prosedur digunakan untuk memanipulasi data.
- Data bersifat pasif.

procedural programming

Dengan teknik ini kita dapat mengatur urutan statemen program, mengembalikan urutan statemen program pada suatu tempat yang kita inginkan . Suatu pemanggilan prosedur digunakan untuk memohon prosedur tersebut berjalan, setelah diproses, arus kendali urutan akan dikembalikan pada urutan proses pemanggilnya.

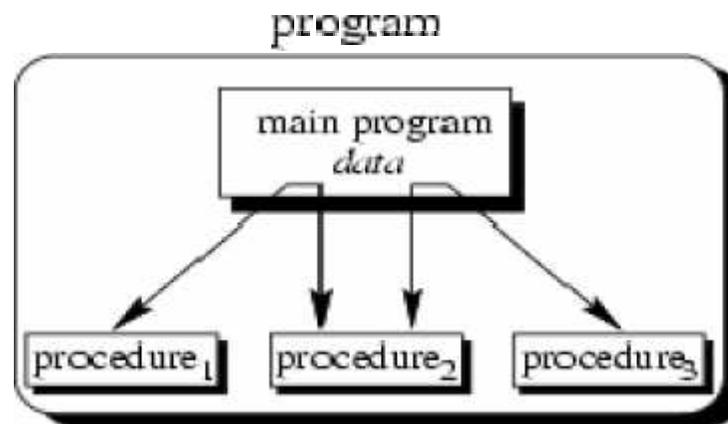
procedural programming

Jika suatu prosedur benar, setiap kali digunakan pasti mengeluarkan hasil benar, dan begitu sebaliknya jika prosedur itu salah juga akan menghasilkan kesalahan jika dipanggil berulang kali.

Tetapi dengan teknik ini kita dapat membatasi tempat pencarian kesalahan, dan jika prosedur yang salah, kita cukup memperbaiki satu kali di baris program dalam prosedur tersebut.

procedural programming

Ada yang menggunakan teknik ini dengan cara program utama hanya berisi baris-baris perintah pemanggil prosedur dan menyediakan data yang akan diolah di prosedur, setelah selesai proses hasilnya akan di bawa ke program utama



Program utama mengkoordinir panggilan ke prosedur dan penyerahan data sesuai parameter.

Modular programming

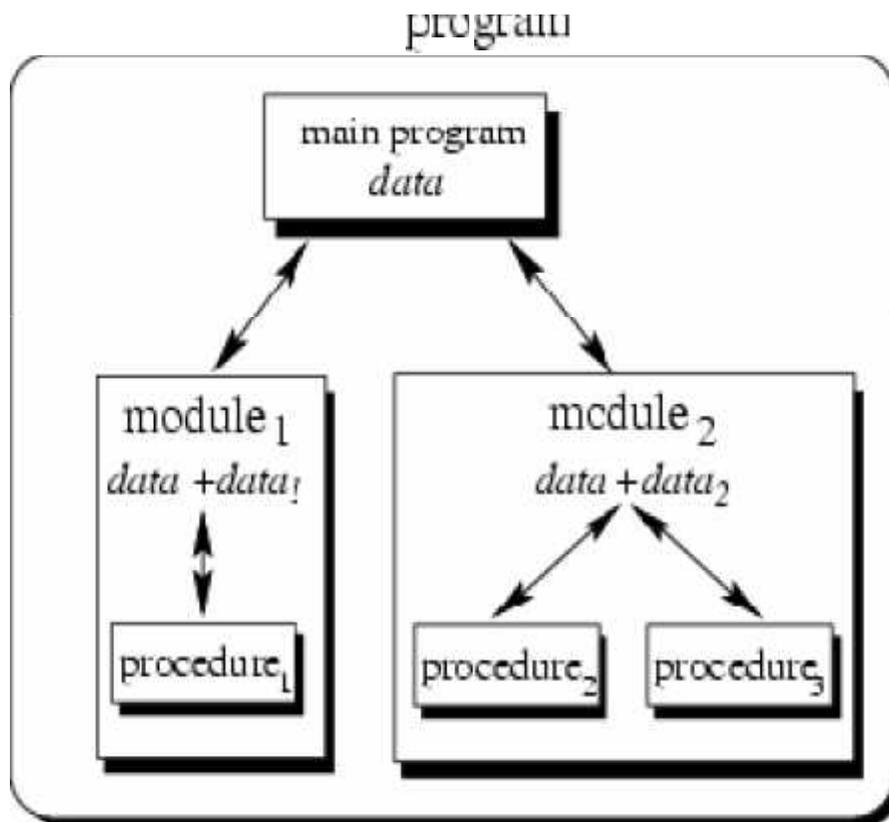
Pada teknik pemrograman modular ini kita mengelompokkan beberapa procedure yang sama fungsinya ke dalam suatu modul yang terpisah dari program utama, modul yang dibuat mungkin lebih dari satu. Dasar penciptaan modul ini, karena programmer menilai prosedur di dalam modul akan banyak digunakan pada program-program yang berlainan.

Modular programming

Masing-masing Modul dapat mempunyai data sendiri dan memungkinkan masing-masing modul untuk mengatur suatu status internal untuk dimodifikasi oleh prosedur yang ada didalamnya.

Jadi, paradigma baru yang ditawarkan pada teknik ini adalah suatu program tidak lagi terdiri dari hanya satu bagian tunggal, tetapi dibagi ke dalam beberapa bagian yang lebih kecil saling berhubungan

modular programming



Program utama mengkoordinir panggilan ke prosedur di dalam modul terpisah dan penyerahan data sesuai parameter.

object-oriented programming

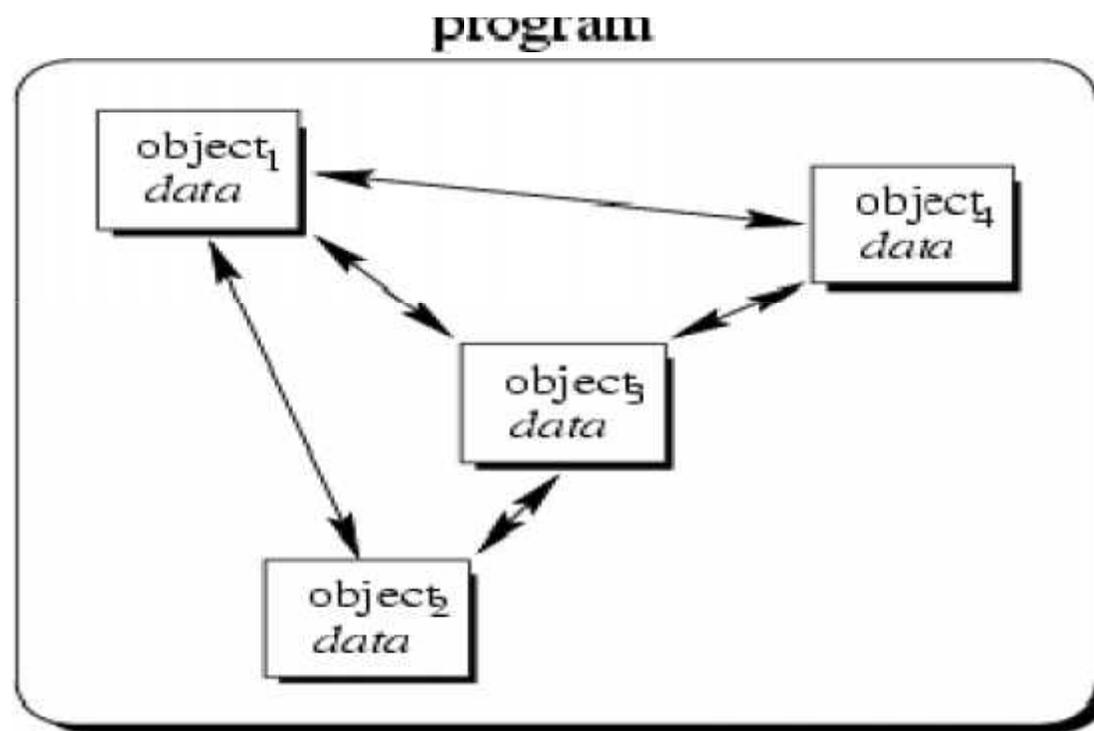
Pada teknik OOP menjadikan kode lebih mudah dipelihara dibandingkan dengan teknik yang lain. OOP menawarkan keistimewaan reuseability, kemampuan yang membuat kode mudah digunakan kembali pada pengembangan aplikasi yang lain, sehingga meningkatkan produktifitas programmer. Pada OOP di program utama terdapat kumpulan objek yang saling berinteraksi.

object-oriented programming

Objek merupakan hal terpenting dalam OOP, contoh mudah pada pemrograman Visual, form dan kotak teks merupakan sebuah obyek yang biasa dilibatkan dalam pembuatan program aplikasi.

Suatu obyek dapat dimanipulasi melalui properti (atribut) yang dimilikinya dan event (kejadian) yang dikenali obyek (contoh : DblClick) serta metode yang dijalankan jika tindakan kejadian dilakukan.

object-oriented programming



Object di dalam program saling berhubungan dengan saling mengirim pesan.

Kata kunci OOP

Objek → dapat berupa Class atau Instances. Harus berasal dari entitas atau konsep dunia nyata.

Class → template untuk membuat obyek.

Atribut → identitas unik dari obyek

Metode → fungsi untuk mengakses atribut atau tugas tertentu

Enkapsulasi → menyembunyikan struktur data dan implementasi suatu class.

Inheritansi → merepresentasikan keterhubungan struktural antar class

Polymorphism → kemampuan untuk merepresentasikan 2 bentuk obyek yang berbeda