

Analisis Kasus/Percabangan (dalam Bahasa C++)

Tim Penyusun Materi PTI-B



KU1072/Pengenalan Teknologi Informasi B
Tahap Tahun Pertama Bersama
Institut Teknologi Bandung





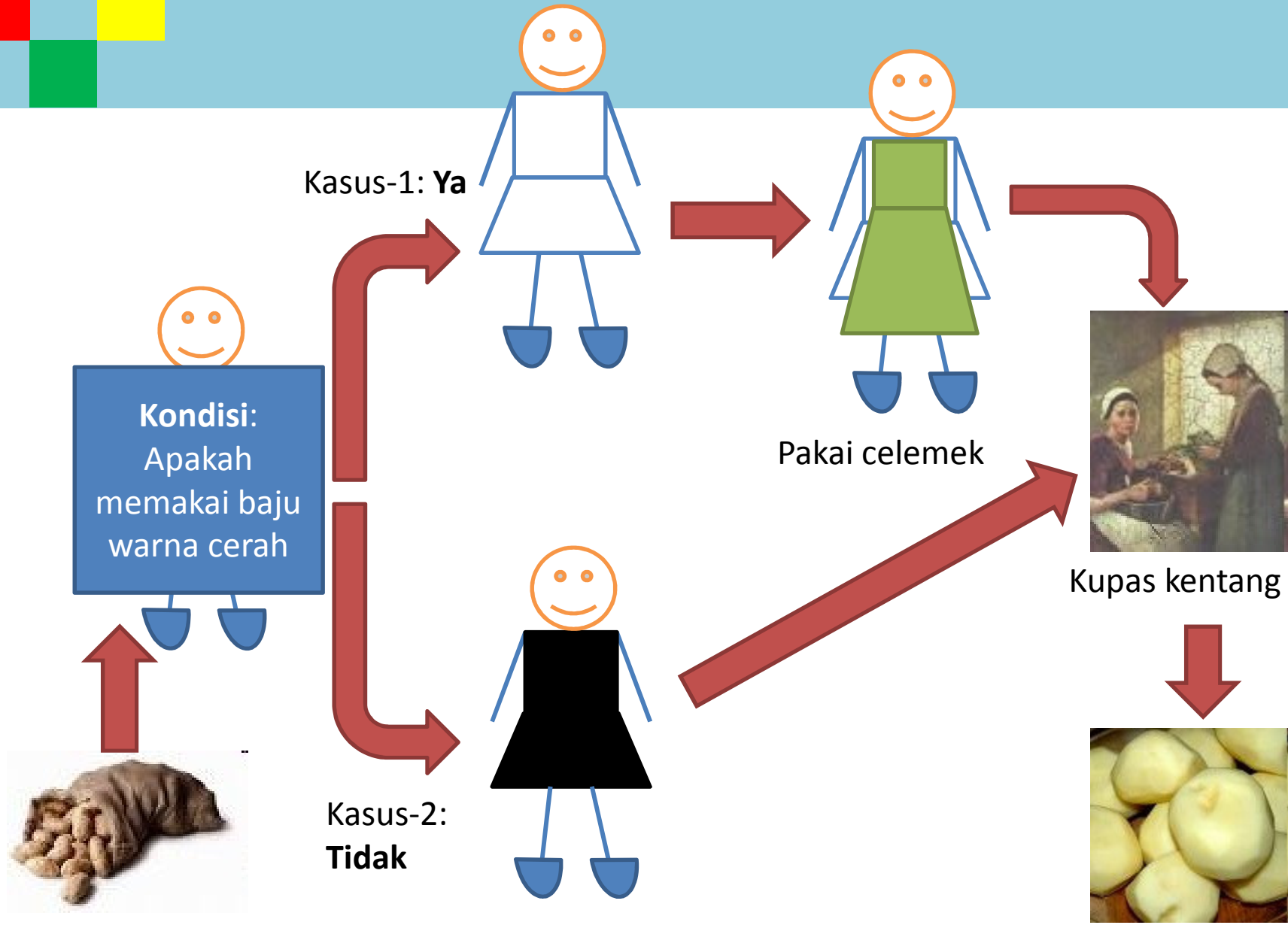
Tujuan

- Mahasiswa memahami pengertian dan jenis-jenis analisis kasus
- Mahasiswa dapat menggunakan notasi analisis kasus dengan benar
- Mahasiswa dapat memanfaatkan jenis-jenis analisis kasus dalam menyelesaikan persoalan sederhana yang diberikan



Mengupas Kentang *Revisited*

- Berdasarkan pengamatan, ada hari-hari di mana ibu **memakai celemek** ketika mengupas kentang, tapi ada hari-hari lain yang tidak
 - Setelah diamati, ternyata jika ibu sedang memakai baju berwarna cerah, maka ibu memakai celemek → takut bajunya terlihat kotor 😊
 - Jika tidak (memakai baju berwarna gelap), maka ibu tidak memakai celemek



Kentang tersedia

20/09/2013

KU1072/Pengenalan Teknologi Informasi B

Kentang terkupas





Contoh lain:

- Diketahui 2 (dua) buah bilangan integer, misalnya A dan B, tentukan manakah yang lebih besar di antaranya keduanya

- **Kondisi** yang digunakan untuk pemeriksaan?

Apakah A lebih besar dari B?

- **Kasus:**

- Jika A lebih besar dari B, maka hasilnya adalah A
- Jika tidak, maka hasilnya B

- Alternatif lain untuk kondisi dan kasus?



Contoh lagi:

- Diberikan sebuah bilangan bulat, misalnya A , nyatakan apakah bilangan tersebut adalah bilangan positif, negatif, atau nol
- **Kondisi: Apakah $A > 0$?**
- **Kasus:**
 - Jika **ya**, maka: A adalah positif
 - Jika **tidak**, maka: **kondisi: Apakah $A = 0$?**
 - Jika **ya**, maka A adalah nol
 - Jika **tidak**, maka A adalah negatif

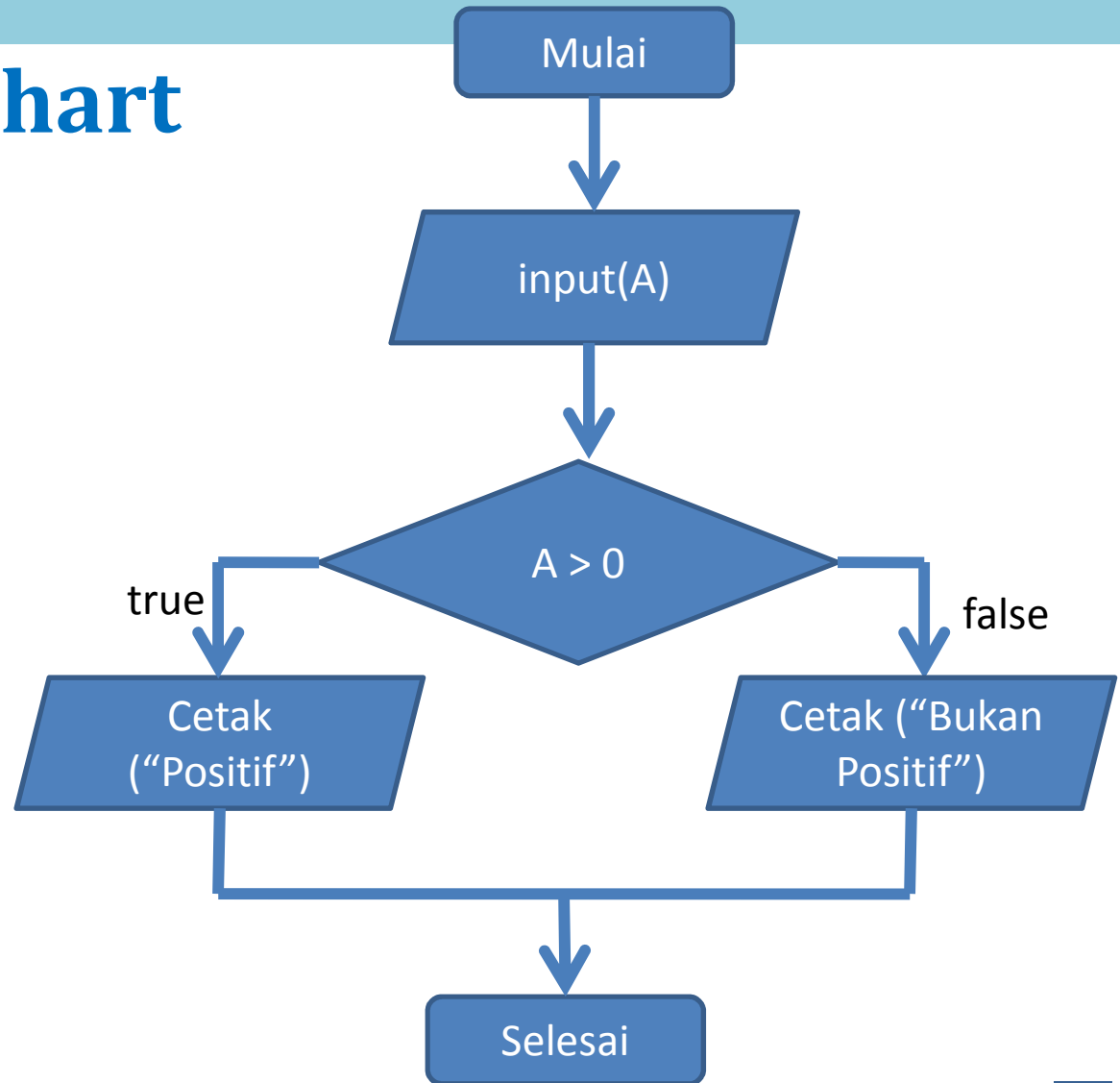


Analisis Kasus

- Memungkinkan kita membuat teks yang sama, namun menghasilkan eksekusi berbeda
- Sering disebut **percabangan**
 - Dari satu langkah ada pilihan (percabangan) ke beberapa langkah
- Terdiri atas:
 - **Kondisi**: ekspresi yang menghasilkan true dan false
 - **Aksi**: statement yang dilaksanakan jika kondisi yang berpasangan dengan aksi dipenuhi

Contoh Flowchart

- Menerima masukan sebuah integer A
- Memeriksa apakah A positif atau bukan
- Mencetak ke layar





Analisis Kasus

- Satu kasus
- Dua kasus komplementer
- Banyak kasus
- Banyak kasus dengan switch



Satu Kasus

```
if ( kondisi ) {  
    aksi;  
}
```

- Kondisi benar → *aksi* dilakukan
- Kondisi tidak benar → tidak terjadi apa-apa



Contoh Satu Kasus

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

int main () {
// KAMUS
    int i;

// ALGORITMA
    cin >> i;
    if (i >= 0) {
        cout << "positif" << endl;
    }
    return 0;
}
```



Dua Kasus Komplementer

```
if (kondisi) {  
    aksi-1;  
}  
else { // not kondisi  
    aksi-2;  
}
```

- Perhatikan kondisi eksplisitnya!

Contoh Dua Kasus Komplementer

```
#include <iostream>
using namespace std;
int main () {
// KAMUS
    int i;
// ALGORITMA
    cin >> i;
    if (i >= 0) {
        cout << "positif";
    }
    else { // i < 0
        cout << "negatif";
    }
    return 0;
}
```



Banyak Kasus

```
if (kondisi-1) {  
    aksi-1;  
}  
else if (kondisi-2) {  
    aksi-2;  
}  
else {  
    ...  
}
```

- (*kondisi-x*) domain harga: [**true**, **false**]
- Property yang perlu diperhatikan untuk (*kondisi-x*) :
disjoint dan *complete*

Contoh Banyak Kasus

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

int main () {
// KAMUS
    int i;
// ALGORITMA
    cin >> i;
    if (i > 0) {
        cout << "positif" << endl;
    }
    else if (i == 0) {
        cout << "nol" << endl;
    }
    else {
        cout << "negatif" << endl;
    }
}
return 0;
}
```



Switch

```
switch (var) {  
    case val-1: aksi-1; break;  
    case val-2: aksi-2; break;  
    ...  
    default: aksi-n; break;  
}
```

- *var* bertipe ordinal, *val-1*, *val-2*, dst adalah constant
- Break untuk keluar dari switch
- Default jika tidak ada yang cocok

Contoh Switch

```
#include <iostream>
using namespace std;
int main () {
// KAMUS
    int i;
// ALGORITMA
    cin >> i;
    switch (i) {
        case 1 : cout << "januari" << endl;
                break;
        case 2 : cout << "februari" << endl;
                break;
        default: cout << "bulan lain" << endl;
                break;
    }
    return 0;
}
```



Analisis Kasus sebagai sarana validasi

```
if ( not kondisi-benar ) {  
    pesan-kesalahan ;  
}  
else { // kondisi-benar  
    aksi;  
}
```

- Digunakan untuk memvalidasi masukan program sehingga masukan yang salah tidak diproses
- *aksi* hanya dilakukan jika *kondisi-benar* terpenuhi, jika tidak maka diberikan *pesan-kesalahan*.



Contoh analisis kasus untuk validasi

- Buatlah program yang digunakan untuk menuliskan ke layar nama-nama bulan dari nomor bulan, yaitu 1 s.d. 12 yang dibaca dari keyboard.
- Program memvalidasi terlebih dahulu apakah masukan dari keyboard sudah benar atau belum, yaitu apakah nomor bulan sudah tepat 1 s.d. 12. Jika tidak tepat, maka dikeluarkan pesan kesalahan yaitu “Masukan nomor bulan tidak tepat”.

Contoh analisis kasus untuk validasi

```
#include <iostream>
using namespace std;
int main () {
// KAMUS
    int nobulan;
// ALGORITMA
    cin >> nobulan;
    if ((nobulan < 1) || (nobulan > 12)) {
        cout << "Masukan nomor bulan tidak tepat" << endl;
        // pesan kesalahan
    } else { // nobulan di antara 1 s.d. 12
        switch (i) {
            case 1 : cout << "januari" << endl; break;
            case 2 : cout << "februari" << endl; break;
            ... // lanjutkan sendiri
        }
    }
    return 0;
}
```



Soal-1

- Buatlah program yang menerima masukan 2 buah integer dan menghasilkan manakah dari kedua integer tersebut yang paling besar.
- Misal integernya adalah A dan B:
 - $A \geq B$ maka A
 - $A < B$ maka B

Alternatif Solusi Soal-1

```
// Program Max2Integer
// Menuliskan nilai terbesar dari masukan 2 integer
#include <iostream>
using namespace std;
int main () {
// KAMUS
    int A, B;

// ALGORITMA
    cin >> A;
    cin >> B;
    if (A >= B) {
        cout << A << endl;
    } else { // A < B
        cout << B << endl;
    }
    return 0;
}
```



Soal-2

- Buatlah sebuah program yang menerima suhu air (dalam derajat celcius) dan menuliskan wujud air ke layar sebagai berikut:
 - Jika suhu air ≤ 0 derajat, maka tuliskan “beku”
 - Jika suhu air > 0 dan kurang dari 100 derajat, maka tuliskan “cair”
 - Jika suhu air ≥ 100 , maka tuliskan “uap”

Alternatif Solusi Soal-2

```
// Program SuhuAir
// Menuliskan wujud air tergantung masukan suhu
#include <iostream>
using namespace std;
int main () {
// KAMUS
    float suhu;

// ALGORITMA
    cin >> suhu;
    if (suhu <= 0) {
        cout << "Beku" << endl;
    } else if ((suhu > 0) && (suhu < 100)) {
        cout << "Cair" << endl;
    } else { // suhu >= 100
        cout << "Uap" << endl;
    }
    return 0;
}
```




Soal-3

- Buatlah program yang menerima 3 buah tahanan dan menghasilkan total tahanan jika dirangkai seri.
- Ketiga masukan tahanan tidak boleh bernilai negatif. Jika salah satu masukan tahanan bernilai negatif, maka diberikan pesan kesalahan “Masukan tahanan tidak boleh negatif”.

Alternatif Solusi Soal-3

```
// Program Tahanan Seri
// Menuliskan total tahanan seri dari 3 buah masukan tahanan
#include <iostream>
using namespace std;
int main () {
// KAMUS
    float R1, R2, R3, Rtotal;

// ALGORITMA
    cin >> R1; cin >> R2; cin >> R3;

    if ((R1 >= 0) && (R2 >= 0) && (R3 >= 0)) {
        Rtotal = R1 + R2 + R3;
        cout << Rtotal << endl;
    } else { // Masukan R1 < 0 atau R2 < 0 atau R3 < 0
        cout << "Masukan tahanan tidak boleh negatif" << endl;
    }
    return 0;
}
```



Soal-4

- Sebuah kalkulator super sederhana **Calc-SS**, digunakan untuk menerima masukan 2 buah bilangan bulat, misalnya A dan B, dan menuliskan beberapa hasil operasi aritmatika terhadap kedua bilangan tersebut, tergantung pilihan operasi melalui character sebagai berikut:
 - Pilihan = a : menuliskan hasil $A + B$
 - Pilihan = b : menuliskan hasil $A - B$
 - Pilihan = c : menuliskan hasil $A * B$
 - Pilihan = d : menuliskan hasil $A/B \rightarrow$ hasilnya real/float
 - Pilihan = e : menuliskan hasil $A \text{ div } B \rightarrow$ hasilnya integer
 - Pilihan = f : menuliskan hasil $A \text{ mod } B$
- Jika dimasukkan pilihan menu character lain, maka dituliskan “Bukan pilihan menu yang benar”

Soal-4: Contoh masukan dan keluaran

Masukan			Keluaran
A	B	pilihan operasi	
3	2	a	$A + B = 5$
3	2	b	$A - B = 1$
3	2	c	$A * B = 6$
3	2	d	$A / B = 1.5$
3	2	e	$A \text{ div } B = 1$
3	2	f	$A \text{ mod } B = 1$
3	2	h	Bukan pilihan menu yang benar

Alternatif Solusi Soal-4

```
//Program Calc-SS
//Menuliskan hasil operasi aritmatika antara 2 integer, tergantung
//pilihan menu
#include <iostream>
using namespace std;
int main () {
// KAMUS
    int A, B;
    char pilihan; //pilihan operasi
// ALGORITMA
    cout << "Masukkan bil pertama (A) = "; cin >> A;
    cout << "Masukkan bil kedua (B) = "; cin >> B;
    cout << "Masukkan pilihan operasi = "; cin >> pilihan;
    switch (pilihan) {
        case 'a' : cout << "A + B = " << A + B << endl; break;
        case 'b' : cout << "A - B = " << A - B << endl; break;
        case 'c' : cout << "A * B = " << A * B << endl; break;
        case 'd' : cout << "A / B = " << (float)A / (float)B << endl;
                    break;
        case 'e' : cout << "A div B = " << A / B << endl; break;
        case 'f' : cout << "A mod B = " << A % B << endl; break;
        default : cout << "Bukan pilihan menu yang benar" << endl;
                    break;
    }
    return 0;
}
```

Type casting
dari int
menjadi float