

Pemrograman Dasar

Pengantar Algoritma



Algoritma

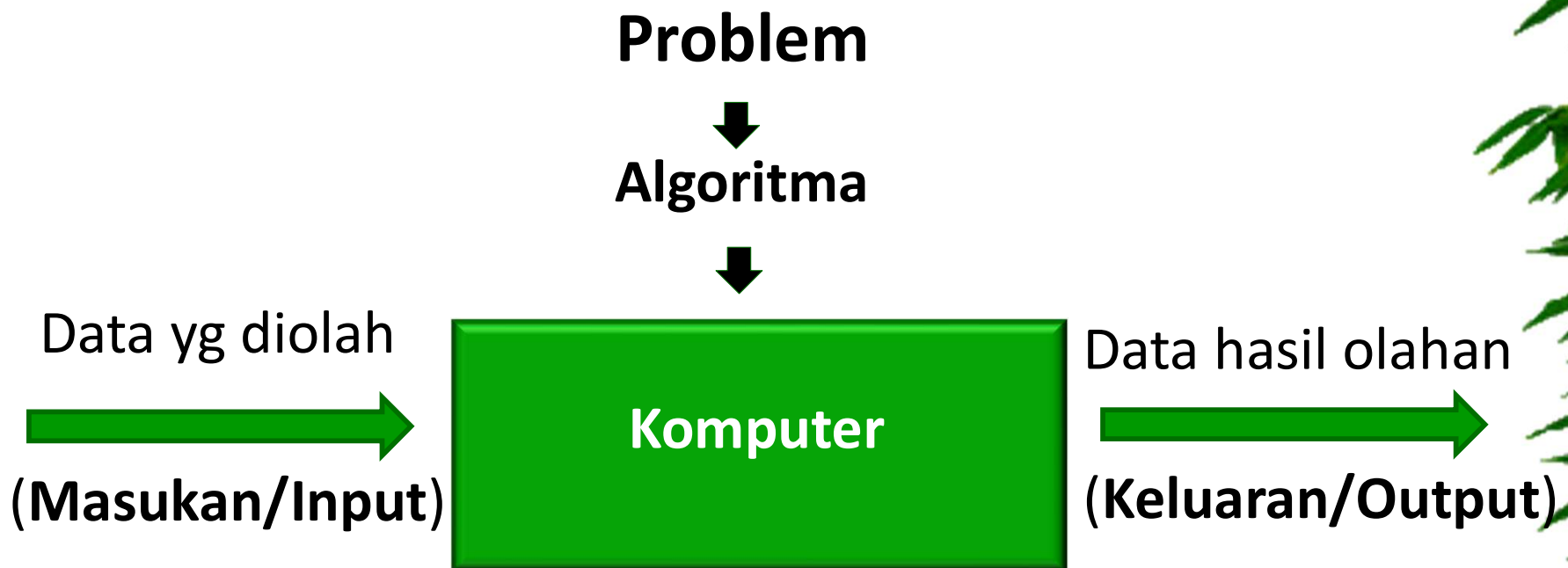
- ★ Diambil dari nama ilmuwan asal Persia **Al Khawarizmi** (Abu Abdallah Muhammad ibn Musa Al Khawarizmi)
 - *Al Khawarizmi: On Calculation with Hindu–Arabic numeral system*, 825M
 - Tulisan di atas awalnya berbahasa Arab, lalu diterjemahkan ke Latin. *Al Khawarizmi* menjadi *Algoritmi*.
 - Berkembang menjadi “algorismus”, yang berarti sistem bilangan desimal.
 - Dalam bahasa Perancis di abad ke-17, algorismus menjadi “algorithm”, kemudian diadopsi dalam bahasa Inggris dengan nama sama.
 - Mulai abad ke-19 istilah ini mulai memiliki arti yang agak berbeda (lihat halaman selanjutnya).



Algoritma

Algoritma adalah sekumpulan instruksi atau langkah-langkah yang jelas (*unambiguous*) dan terbatas untuk mencari solusi suatu masalah.

- Untuk mendapatkan keluaran yang dibutuhkan dari masukan yang sah dalam waktu yang terbatas
- Algoritma sebenarnya cara, bukan hasil atau solusi



Algoritma

- ★ Di pemrograman komputer, algoritma diimplementasikan ke dalam **program** komputer, yaitu:
 - satu set instruksi atau langkah-langkah yang dijalankan dengan komputer untuk menyelesaikan suatu masalah.

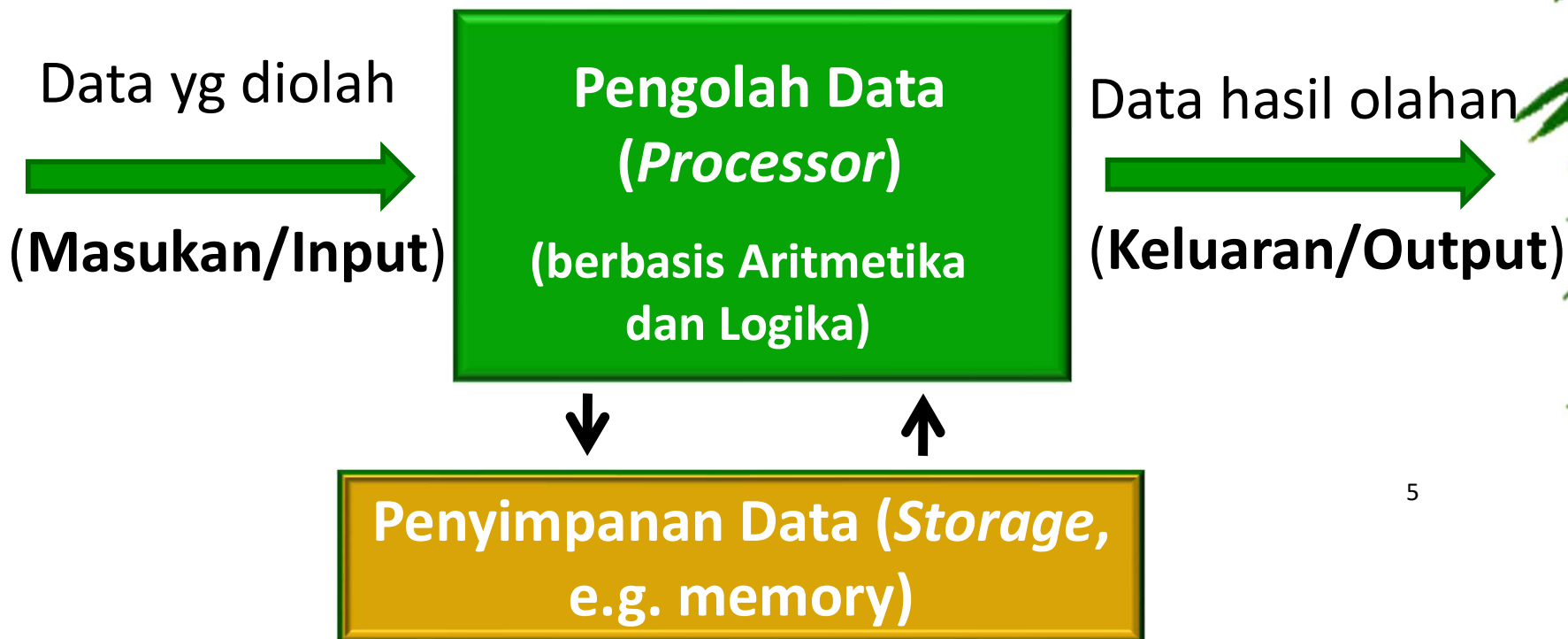


Pemodelan Komputer Sederhana

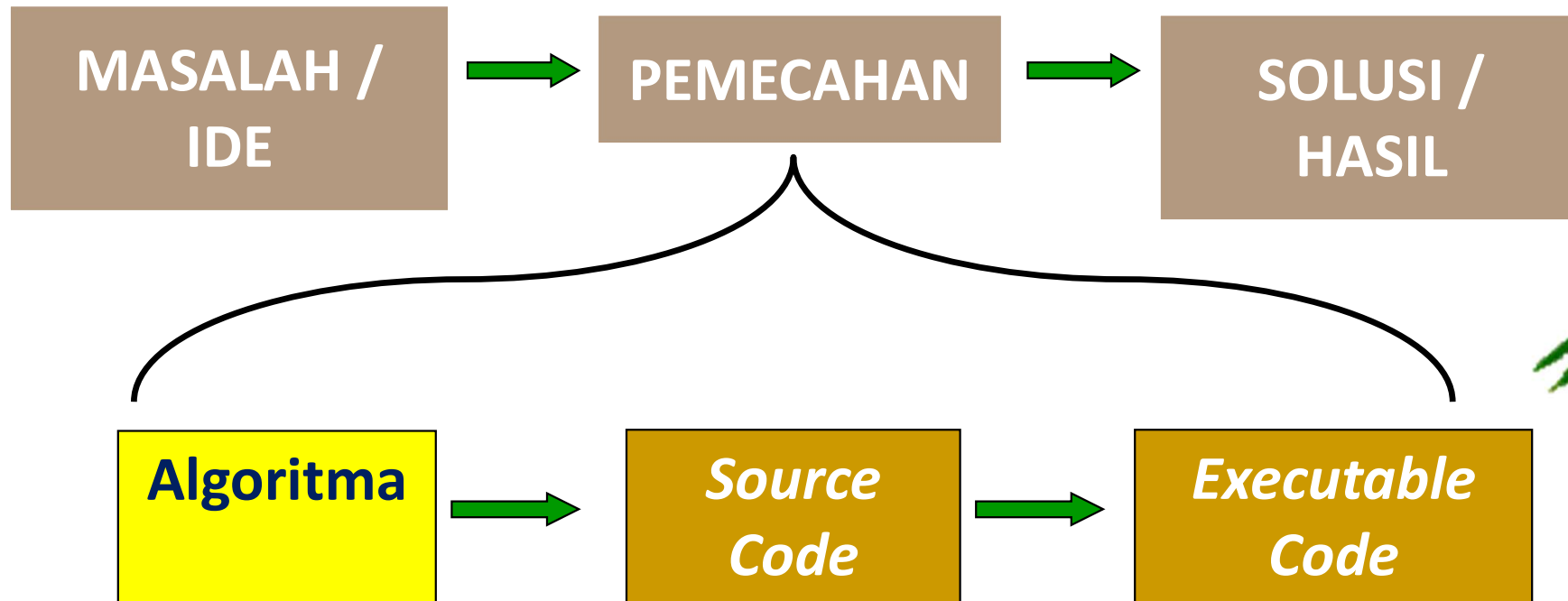
COMPUTER → TO – COMPUTE + ER

↓
menghitung/mengolah bilangan

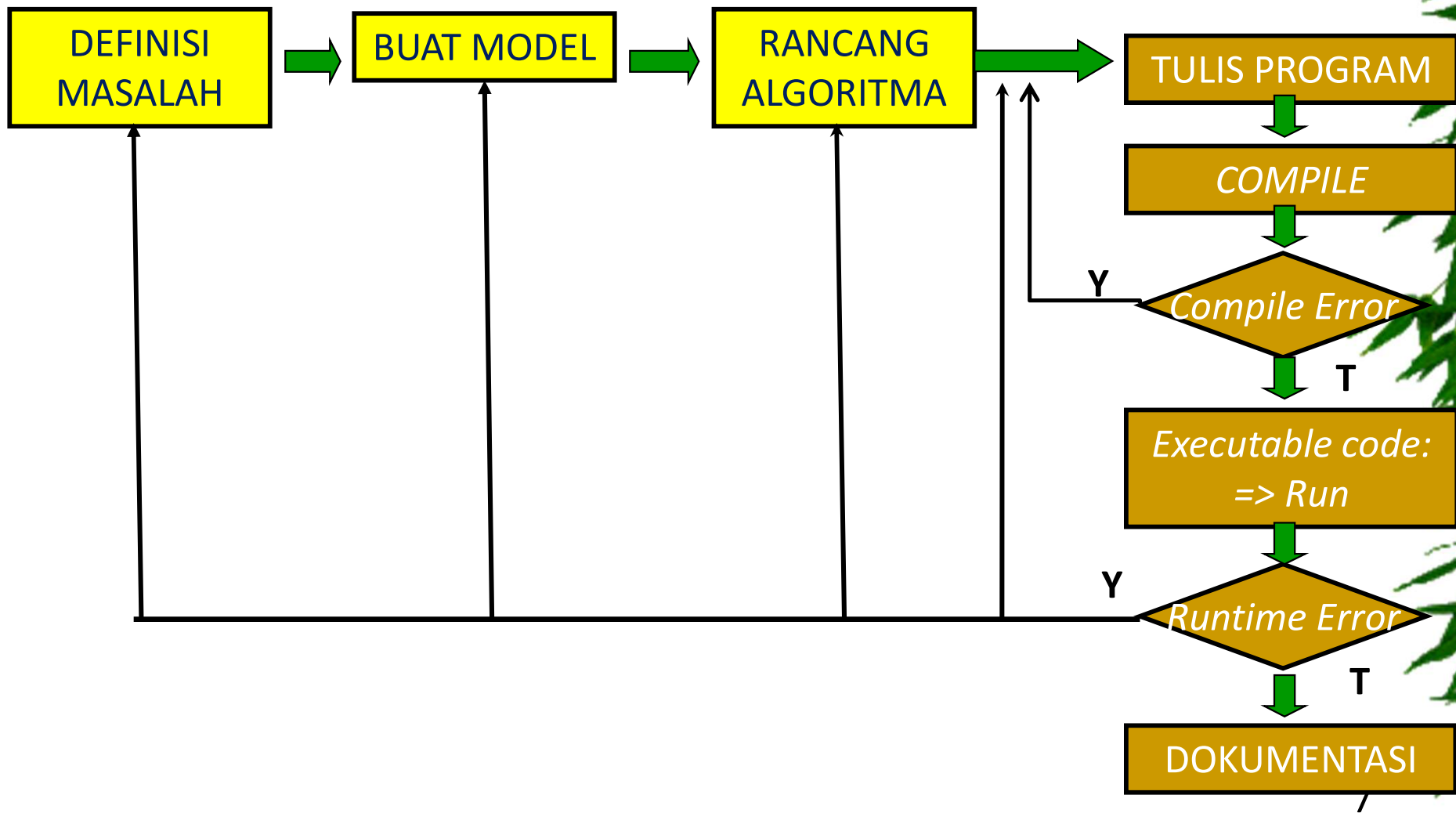
↓
mengolah data



Tahap Pengembangan Algoritma



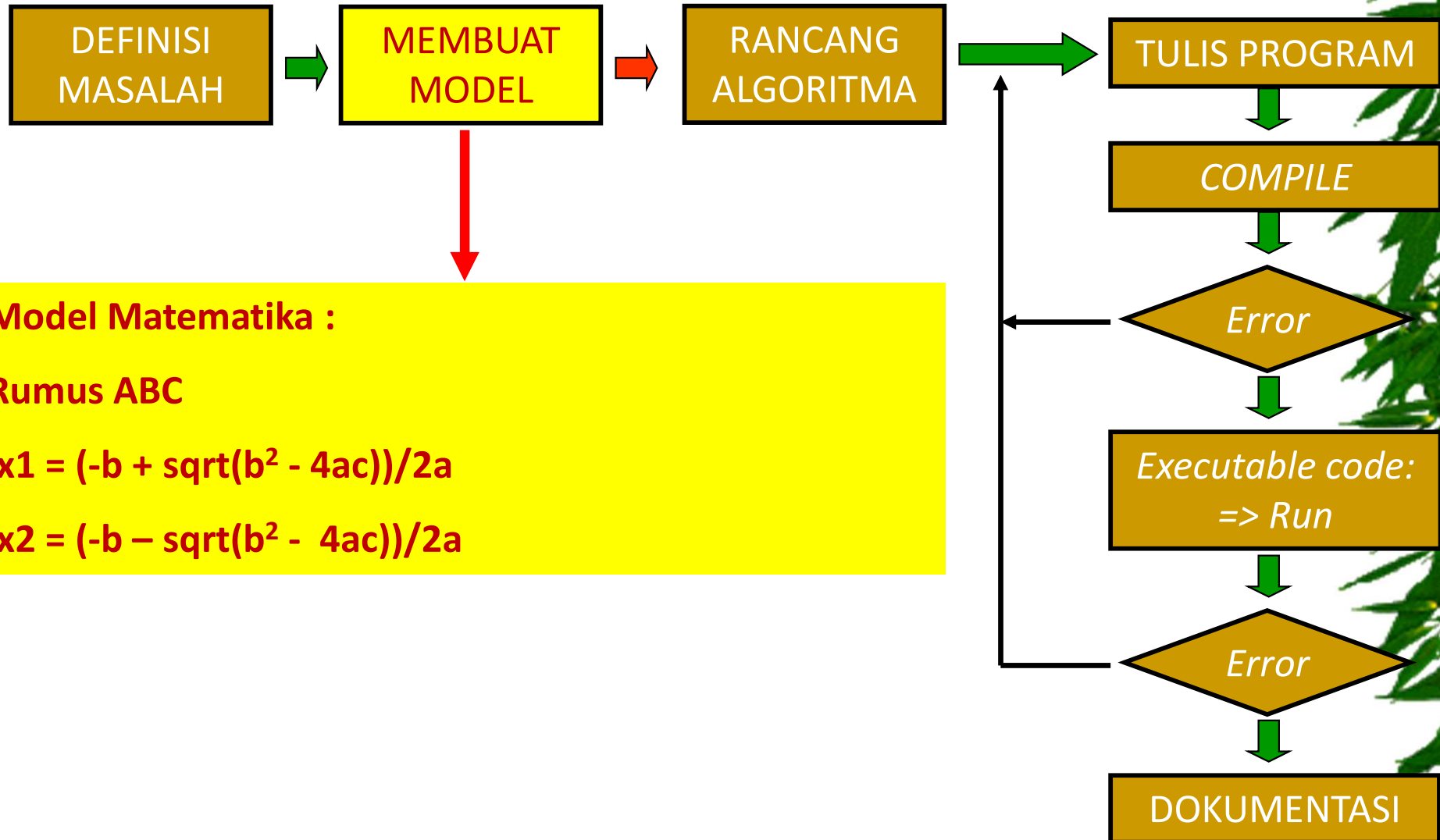
Tahap Pengembangan Algoritma



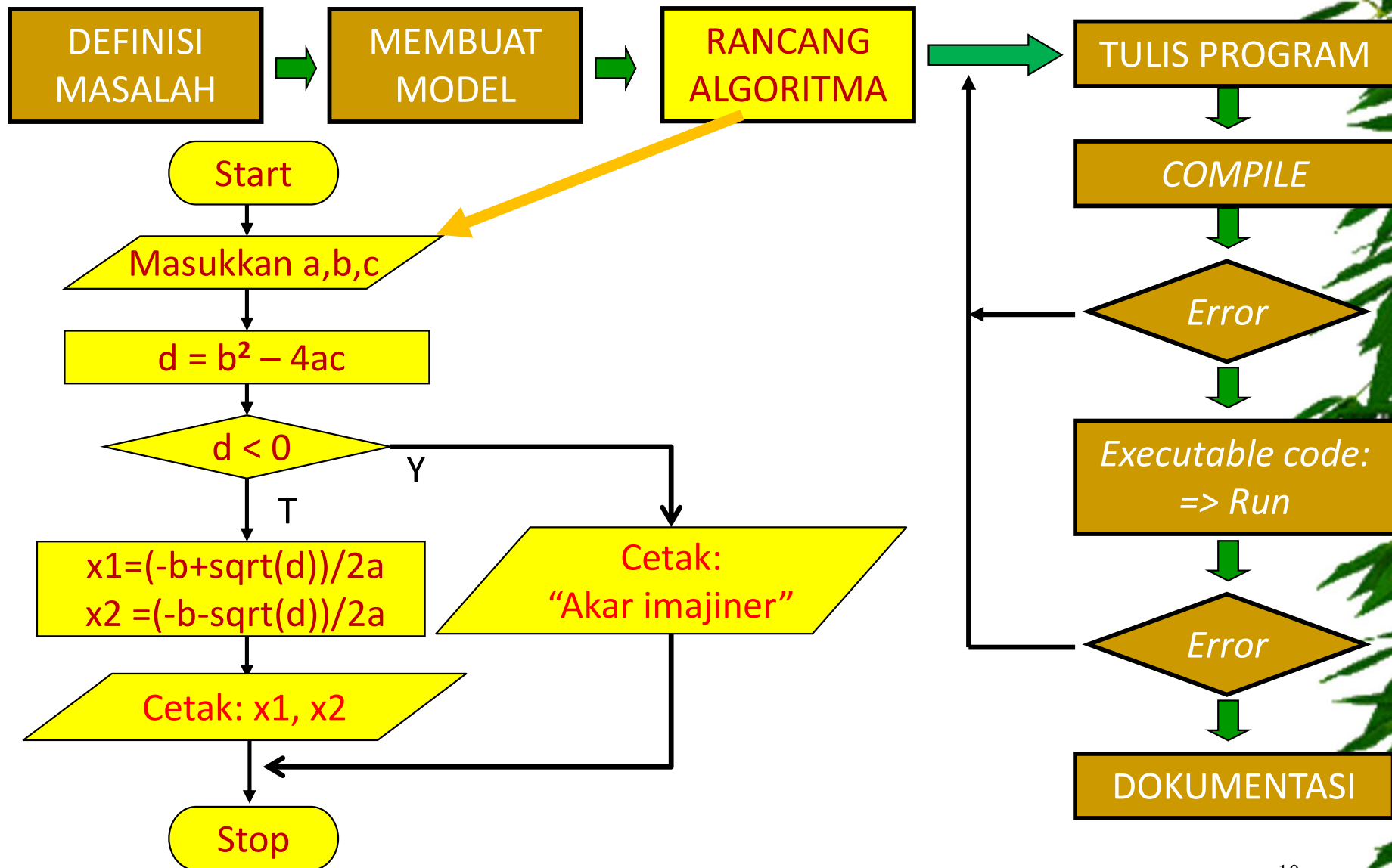
Tahap Pengembangan Algoritma



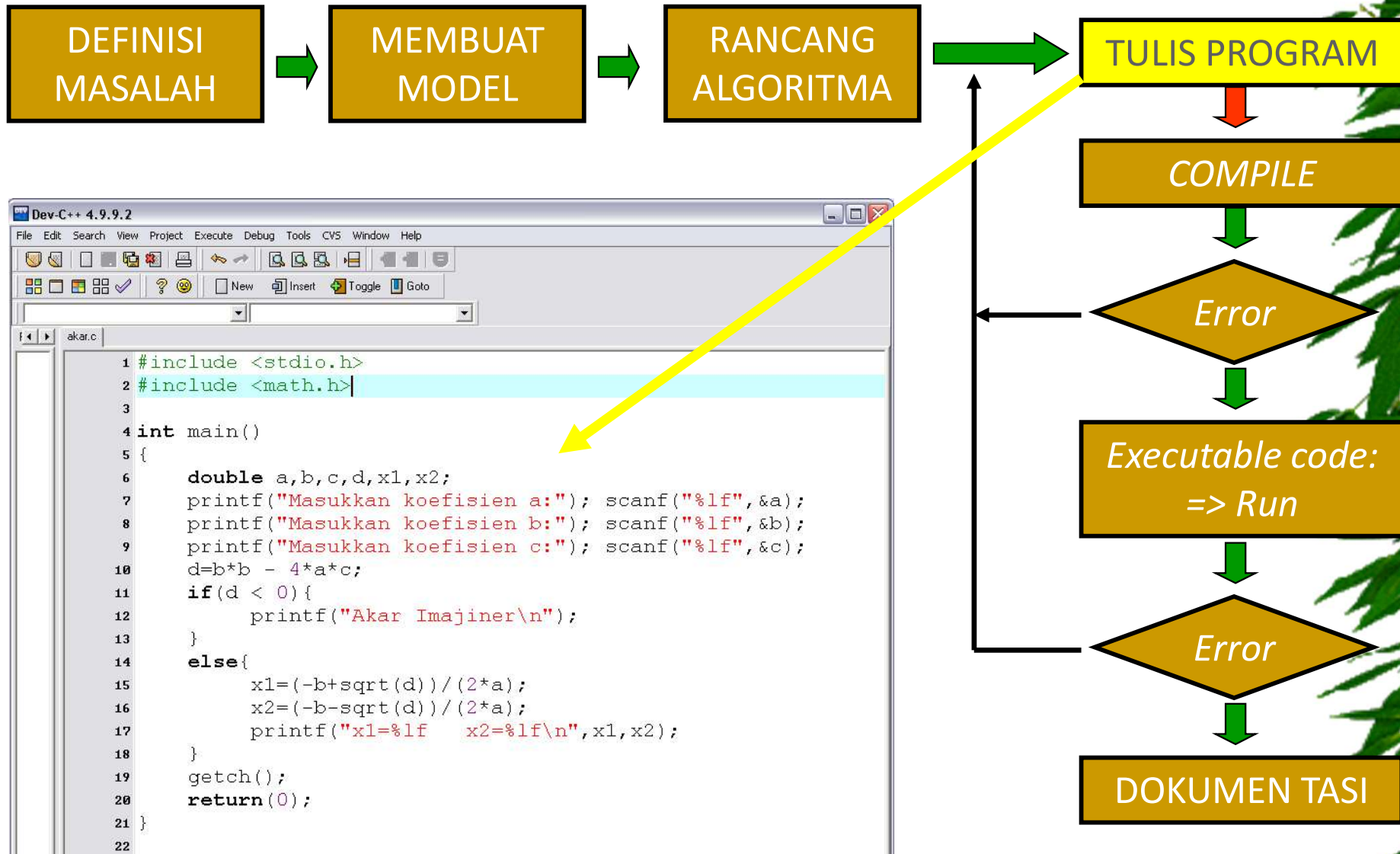
Tahap Pengembangan Algoritma



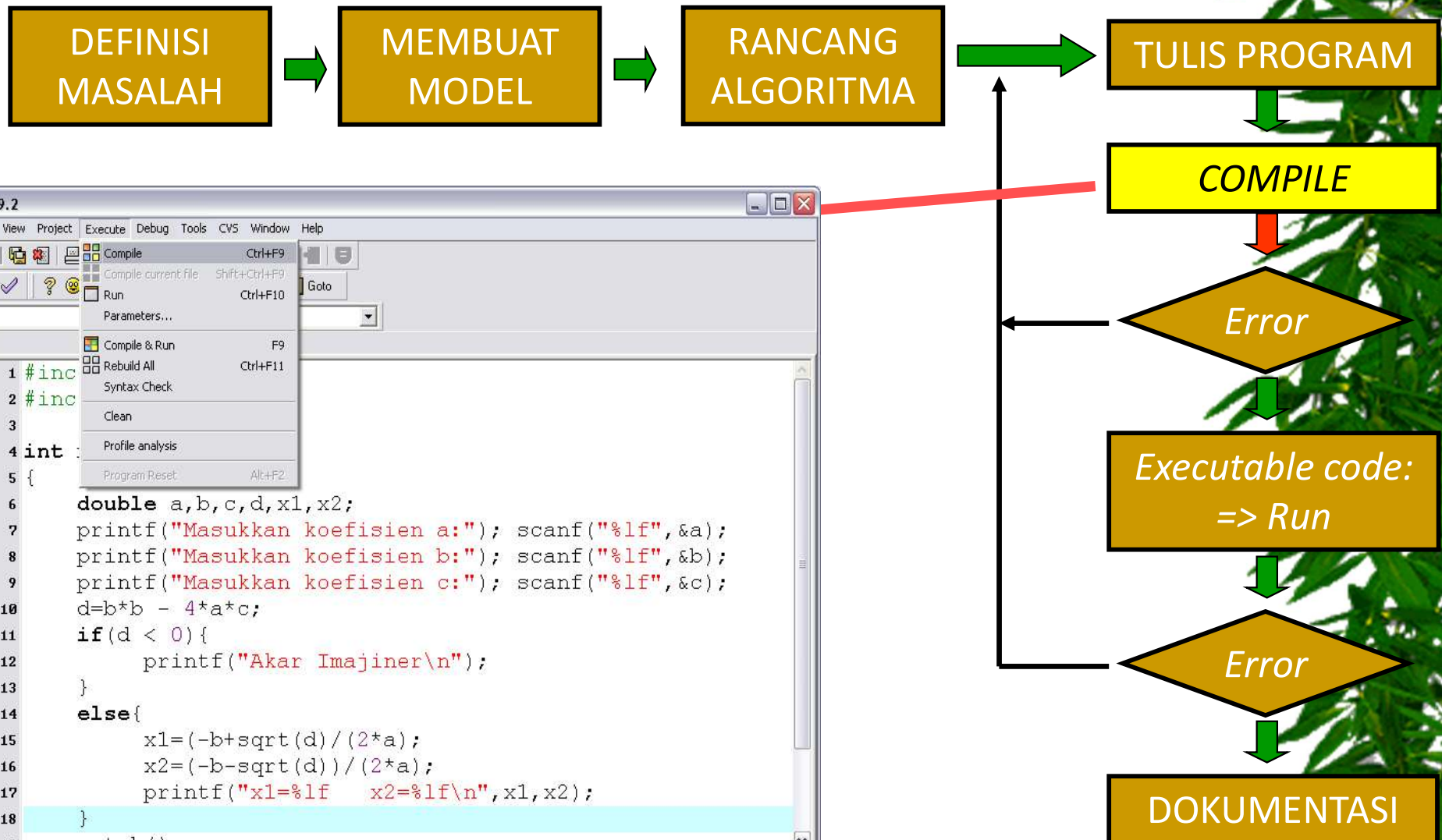
Tahap Pengembangan Algoritma



Tahap Pengembangan Algoritma



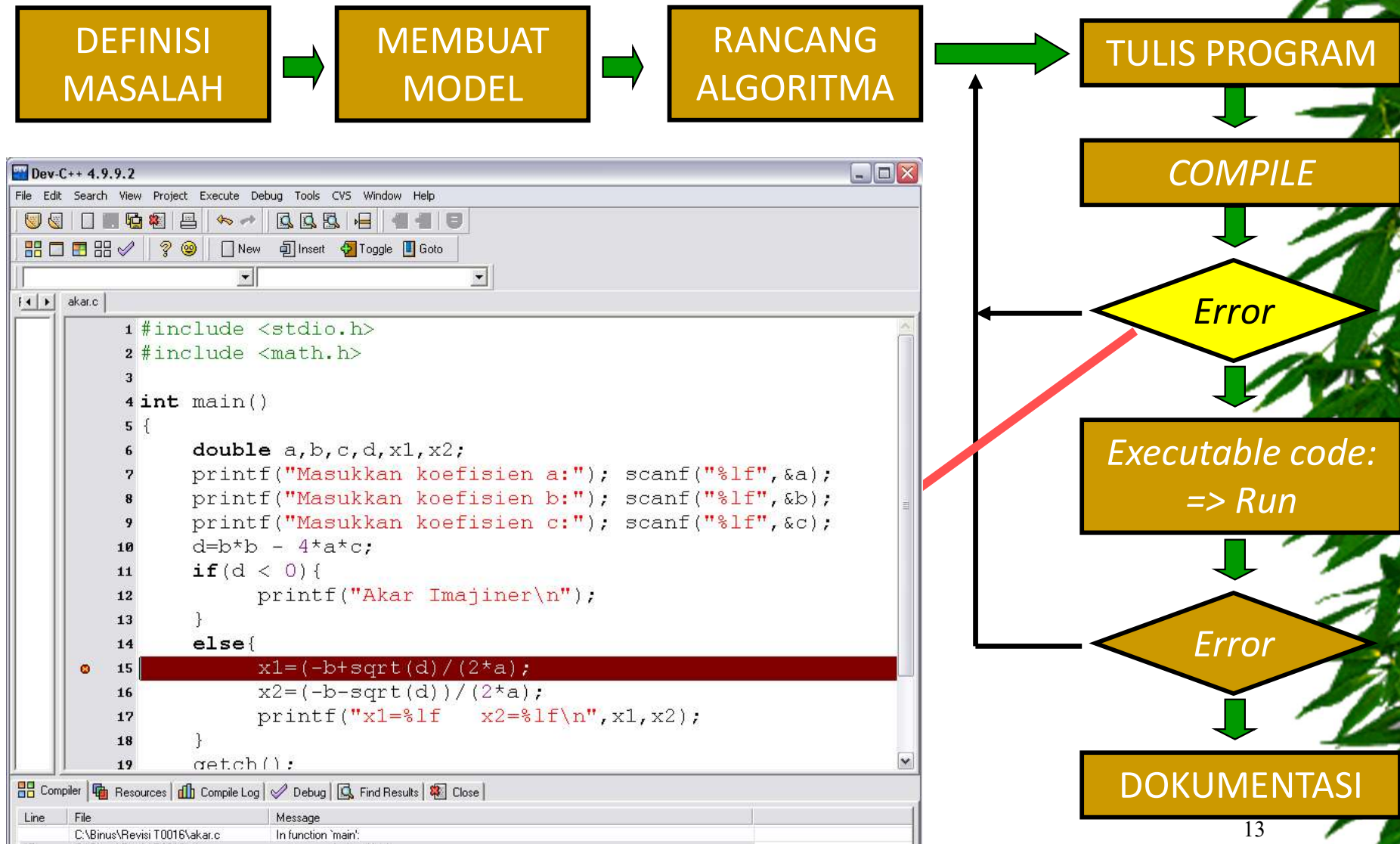
Tahap Pengembangan Algoritma



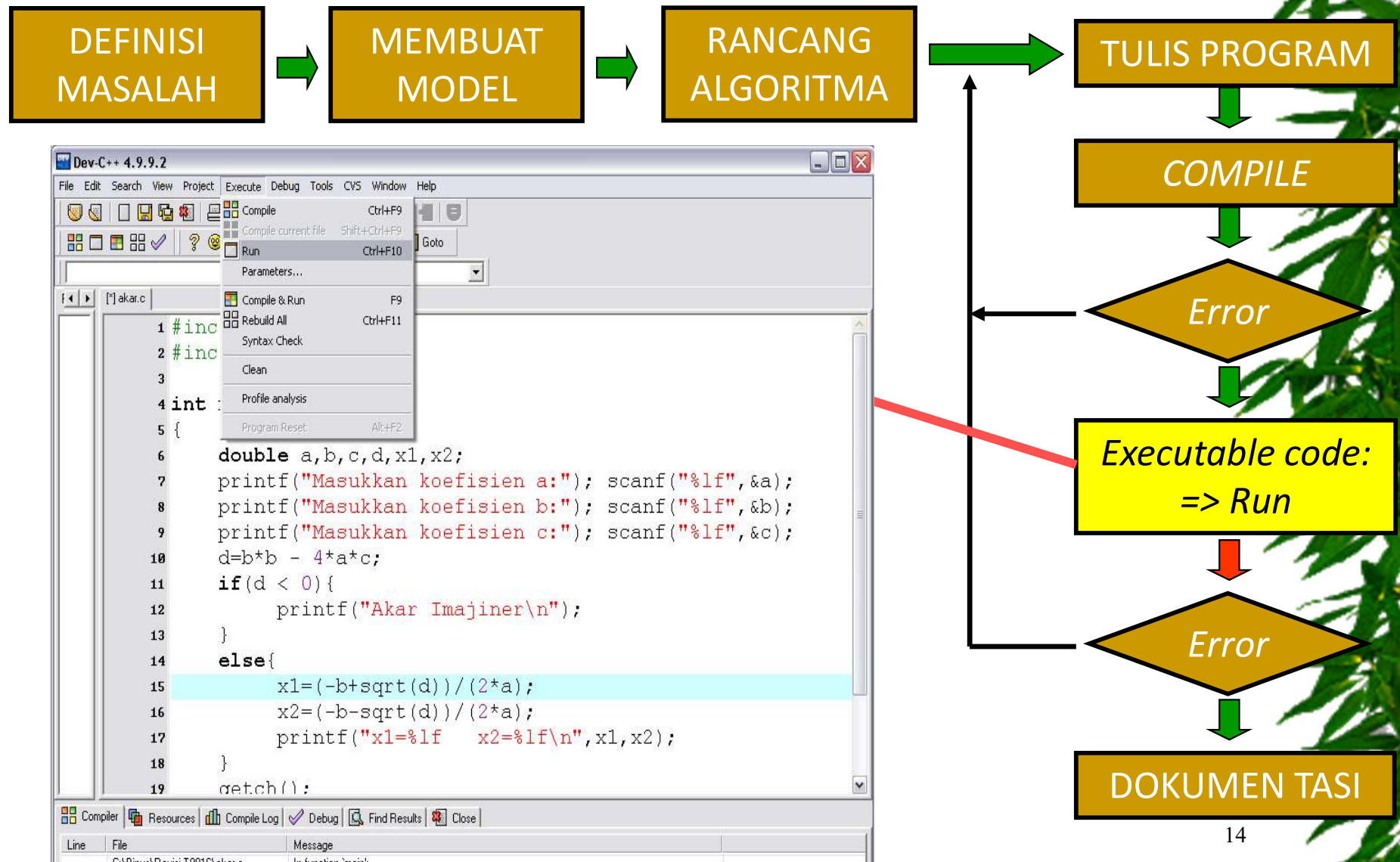
```
#include <stdio.h>
#include <math.h>

int main()
{
    double a,b,c,d,x1,x2;
    printf("Masukkan koefisien a:"); scanf("%lf",&a);
    printf("Masukkan koefisien b:"); scanf("%lf",&b);
    printf("Masukkan koefisien c:"); scanf("%lf",&c);
    d=b*b - 4*a*c;
    if(d < 0){
        printf("Akar Imajiner\n");
    }
    else{
        x1=(-b+sqrt(d)/(2*a);
        x2=(-b-sqrt(d)/(2*a);
        printf("x1=%lf x2=%lf\n",x1,x2);
    }
    getch();
}
```

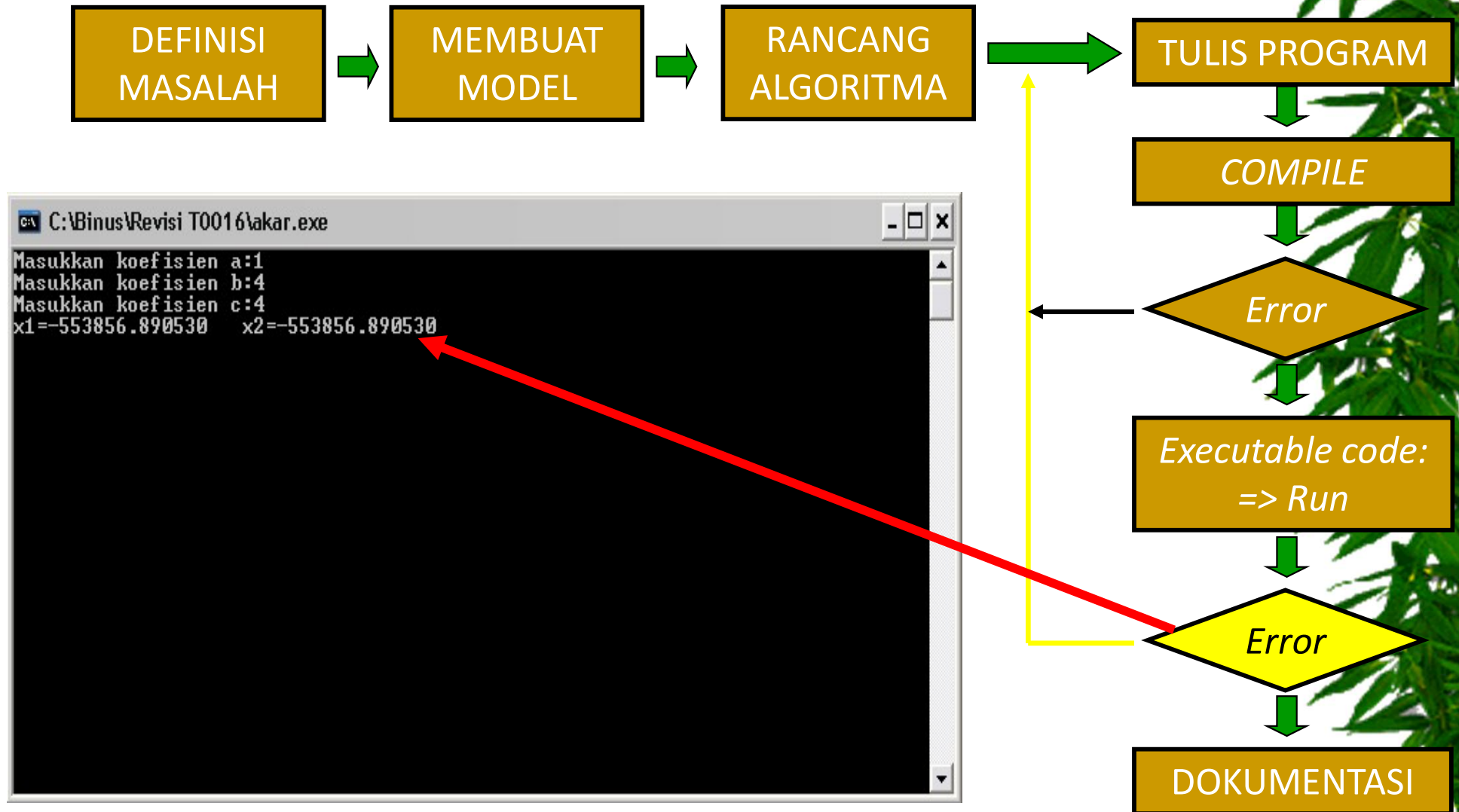

Tahap Pengembangan Algoritma



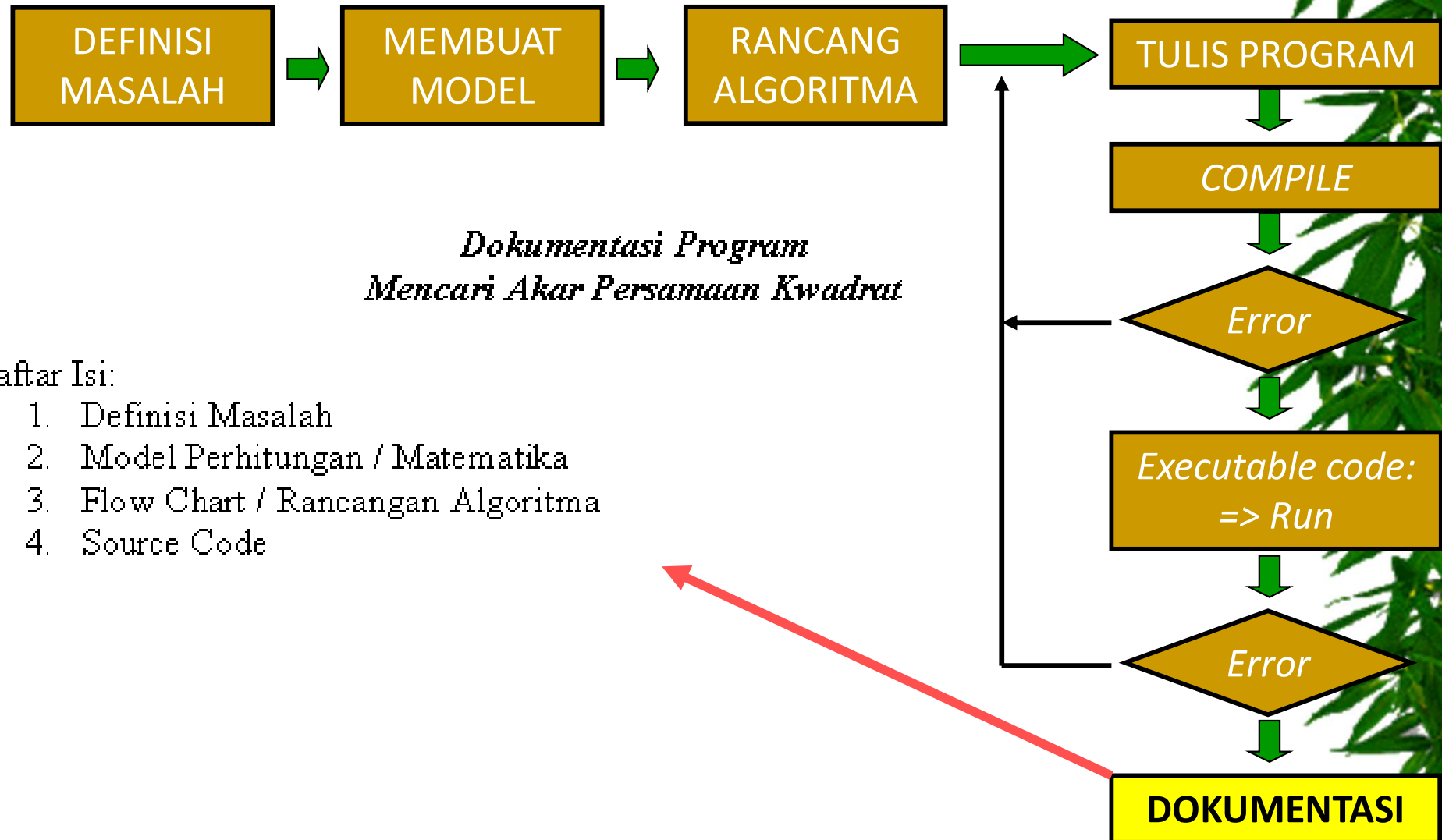
Tahap Pengembangan Algoritma



Tahap Pengembangan Algoritma



Tahap Pengembangan Algoritma



Penyajian Algoritma

- ★ Algoritma dapat diekspresikan dalam bentuk:
 - Tulisan, mis.: *structured English, pseudocode, notasi lain*
 - Gambar, mis.: *flow chart, activity diagram*



Pseudocode

- ★ Kode atau bahasa informal yang digunakan untuk mendeskripsikan algoritma atau operasi-operasi dari sebuah program komputer
- ★ Biasanya menggunakan bahasa yang dekat dengan sebuah bahasa pemrograman
- ★ Ditulis dalam bahasa natural sederhana (mis. Inggris atau Indonesia) atau bahasa matematika
- ★ Kata kunci (*keyword*) digunakan untuk menjelaskan **struktur kendali** (misalnya: “jika”, “ulangi”, “sampai”, “if”, “repeat”, “until”)

Contoh pseudocode

Algoritma Berangkat Kuliah

Mulai

Bangun dari Tempat Tidur

Sholat/Doa Pagi

Mandi Pagi

Sarapan Pagi

Pergi Ke Kampus

Cari Ruang Kuliah

Masuk Kelas

Mengikuti Kuliah

Selesai



Contoh pseudocode

Algoritma Berangkat Kuliah 2

Mulai

Bangun dari Tempat Tidur

Sholat/Doa Pagi

Jika waktu persiapan masih ada

Mandi Pagi

Sarapan Pagi

Pergi Ke Kampus

Cari Ruang Kuliah

Masuk Kelas

Mengikuti Kuliah

Selesai



Contoh pseudocode

Algoritma Berangkat Kuliah 3

Mulai

Bangun dari Tempat Tidur

Sholat/Doa Pagi

Jika Hari ini adalah hari libur

Tidur lagi

tetapi jika Hari ini bukan hari libur

Mandi Pagi

Sarapan Pagi

Pergi Ke Kampus

Cari Ruang Kuliah

Masuk Kelas

Mengikuti Kuliah

Selesai

Contoh pseudocode

Algoritma Sarapan Pagi

Mulai

Ambil piring

Masukkan nasi dan lauk dalam piring

Ambil sendok dan garpu

Ulangi

Angkat sendok dan garpu

Ambil nasi dan lauk

Suapkan ke dalam mulut

Taruh sendok dan garpu

Kunyah

Sampai (nasi dan lauk habis) ATAU kekenyangan

Bereskan piring, sendok dan garpu

Selesai



Contoh pseudocode

Algoritma Menggunakan Kalkulator

Mulai

Nyalakan kalkulator

Kosongkan memori kalkulator

Ulangi

Input harga

Tekan tombol Plus (+)

Sampai semua harga diinput

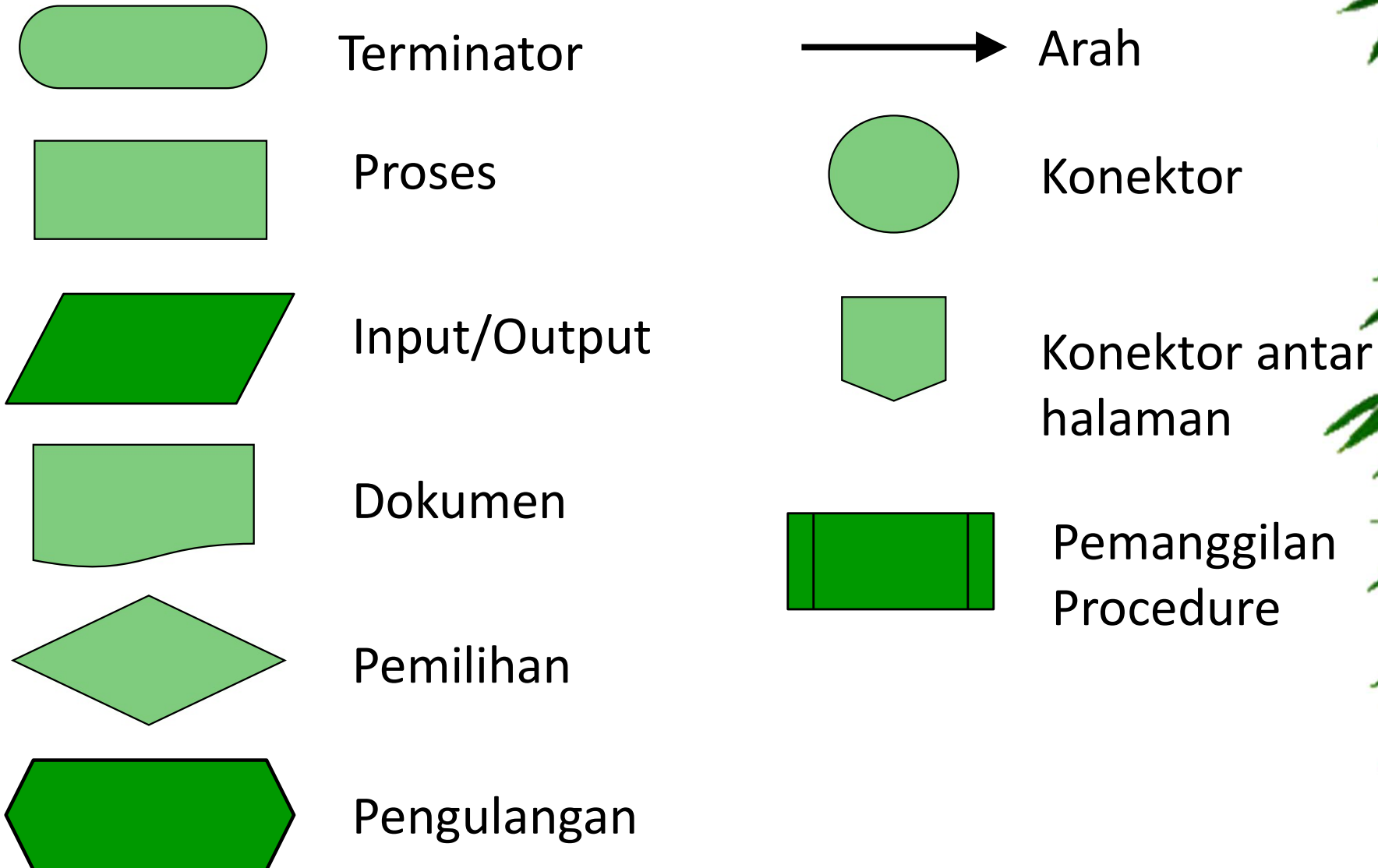
Tampilkan total harga

Matikan kalkulator

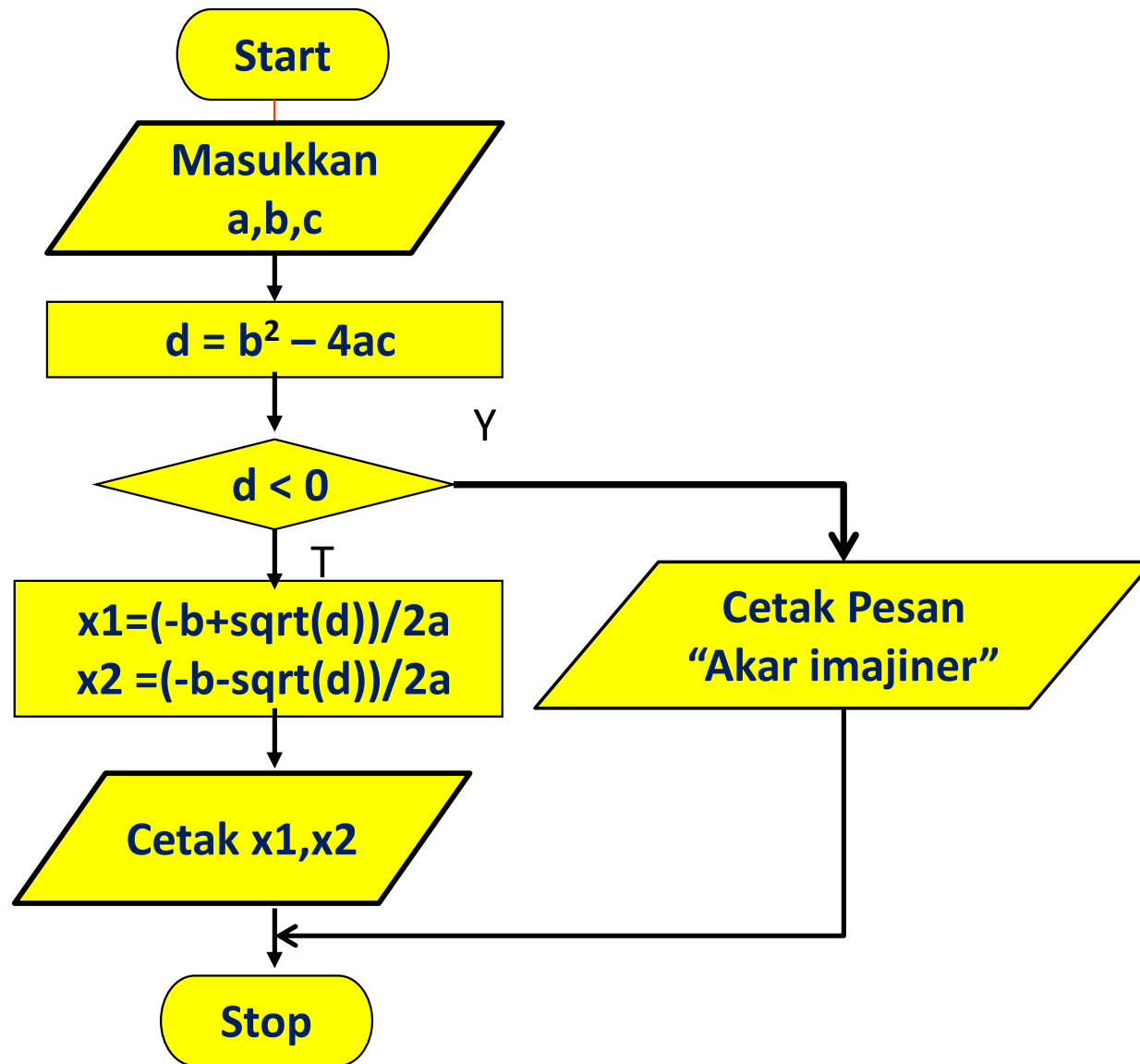
Selesai



Diagram Alir (*Flow Chart*)



Contoh Diagram Alir



Kriteria algoritma yang baik

★ *Correctness* (kebenaran)

- Menghasilkan keluaran yang benar untuk masukan yang valid dalam waktu yang terbatas
- Mempunyai logika yang benar untuk memecahkan masalah.

★ *Simplicity* (kesederhanaan)

- Mudah dipahami, mudah diprogram
- “Indah”

★ *Efficiency* (efisiensi)

- *Time efficiency* (efisiensi waktu): seberapa cepat
- *Space efficiency* (efisiensi ruang): seberapa banyak memori yang dibutuhkan



Kriteria algoritma yang baik

- ✧ *Generality* (keumuman)
- ✧ *Lain-lain*
 - Ditulis dengan bahasa baku terstruktur sehingga tidak menimbulkan arti ganda atau ambigu, dan mudah diimplementasikan kedalam bahasa pemrograman.



PR

- ★ Ubahlah algoritma yang tertulis dalam *pseudocode* pada contoh-contoh di *slides* kuliah Pengantar Algoritma ini ke dalam diagram alir (*flowchart*). Algoritma tersebut:
 - Algoritma berangkat kuliah
 - Algoritma berangkat kuliah 2
 - Algoritma berangkat kuliah 3
 - Algoritma sarapan pagi
 - Algoritma menggunakan kalkulator

