

BAB 1

Pengantar Algoritma dan Pemrograman

Tujuan Pembelajaran

1. Mahasiswa mampu menjelaskan pengertian program komputer
2. Mahasiswa mampu menjelaskan pengertian algoritma
3. Mahasiswa mampu menjelaskan perbedaan penyajian algoritma dengan bahasa dan penyajian algoritma dengan *flowchart*
4. Mahasiswa mampu menyusun algoritma dan menyajikannya dalam bentuk bahasa maupun *flowchart*
5. Mahasiswa mampu menjelaskan langkah-langkah penyusunan program komputer
6. Mahasiswa mampu menjelaskan kriteria program komputer yang baik

A. Sistem Komputer

Komputer adalah alat yang dipakai untuk mengolah data menurut prosedur yang telah dirumuskan. Kata *computer* pada awalnya dipergunakan untuk menggambarkan orang yang perkerjaannya melakukan perhitungan aritmetika,

dengan atau tanpa alat bantu, tetapi arti kata ini kemudian dipindahkan kepada mesin itu sendiri. Asal mulanya, pengolahan informasi hampir eksklusif berhubungan dengan masalah aritmetika, tetapi komputer modern dipakai untuk banyak tugas yang tidak berhubungan dengan matematika. Komputer dapat membantu manusia dalam pekerjaan sehari-harinya seperti: pengolahan kata, pengolahan angka, dan pengolahan gambar.

Sistem komputer adalah suatu jaringan elektronik yang terdiri dari perangkat lunak dan perangkat keras yang melakukan tugas tertentu (menerima input, memproses input, menyimpan perintah-perintah, dan menyediakan output dalam bentuk informasi). Selain itu dapat pula diartikan sebagai elemen-elemen yang terkait untuk menjalankan suatu aktivitas dengan menggunakan komputer.

Penggunaan sistem komputer melibatkan beberapa elemen penting yang terdiri dari perangkat keras (*hardware*), perangkat lunak (*software*) dan manusianya (*brainware*). Komponen-komponen tersebut merupakan elemen yang terlibat dalam suatu sistem komputer.

1. Hardware

Hardware atau perangkat keras adalah semua bagian fisik yang ada pada komputer, bisa dilihat secara kasat mata dan mampu disentuh secara fisik. *Hardware* digunakan untuk mendukung kinerja perangkat komputer dan setidaknya ada 3 fungsi *hardware* yang harus diketahui, antara lain:

- a. Sebagai media input data
- b. Sebagai pengolah data
- c. Sebagai penampil output (luaran)

Hardware komputer dapat dibagi menjadi dua bagian, yaitu *hardware* utama dan *hardware* pendukung.

Hardware utama meliputi:

- ✓ **CPU** sebagai unit pengolah data,
- ✓ **RAM** sebagai tempat penyimpanan data sementara,
- ✓ **Hardisk** sebagai tempat penyimpanan data semi permanen,
- ✓ **Keyboard** sebagai alat penginput data,
- ✓ **Monitor** sebagai alat penampil output dalam bentuk visual,
- ✓ **Speaker** sebagai alat penampil output dalam bentuk audio.

Adapun *hardware* pendukung dapat berupa:

- ✓ Mouse
- ✓ Printer
- ✓ Scanner
- ✓ Modem
- ✓ Flash disk.

2. Software

Software atau perangkat lunak adalah perintah dalam bentuk rekaman yang berfungsi mengendalikan kerja perangkat keras komputer. *Software* dapat berupa program aplikasi yang dibuat oleh developer untuk dapat mengoperasikan atau menjalankan suatu perintah. Secara tidak langsung *software* ini pun bisa dibiladapat dikatakan sebagai perangkat yang menggerakkan *hardware* agar dapat bekerja dengan baik. Perangkat keras komputer bekerja dengan menerima sinyal selektronik digital dengan nilai *hight* (1) atau *low* (0).

Beberapa fungsi utama dari *software*, antara lain:

- ✓ Sebagai sistem operasi atau sistem pendukung

- ✓ Untuk mengoperasikan hardware
- ✓ Sebagai penerjemah setiap perintah atau instruksi ke dalam bahasa pemrograman sehingga bisa diterima oleh hardware dengan baik

Secara umum, *software* dapat dikelompokkan dalam 3 bagian, yaitu

- a. *Software system operasi (operating system)*, yaitu program yang ditulis untuk mengendalikan dan mengoordinasikan kegiatan dari system computer. Contoh : windows, linux.
- b. *Software bahasa (Bahasa pemrograman)*, yaitu program yang digunakan untuk menerjemahkan perintah-perintah yang ditulis dalam bahasa pemrograman ke dalam bahasa mesin (level *high/1* atau level *low/0*), agar dapat dipahami oleh komputer. Contoh : Turbo Pascal, Delphi, C, Visual C, Visual Basic, MATLAB, dan lain-lain.
- c. *Software aplikasi*, yaitu program yang ditulis menggunakan *software* bahasa, seperti microsoft word, microsoft excel, microsoft power point, microsoft access, spss, dan sejenisnya.

3. Brainware

Brainware adalah orang yang mengendalikan komponen perangkat pada komputer. Meski dilengkapi dengan *hardware* dan *software* canggih, komputer tetap tidak akan dapat beroperasi tanpa adanya *brainware*. Seseorang yang mengoperasikan atau mengatur sistem di dalam komputer, itulah yang disebut dengan *brainware*. Contoh-contoh *brainware* meliputi:

- ✓ **User/Operator**, yaitu orang yang memiliki tugas mengoperasikan komputer.

- ✓ **Administrator**, dapat disebut sebagai *brainware*, karena biasanya mampu mengatur sistem kerja, urutan, dan pengelolaan data sampai pada *output*.
- ✓ **Programmer**, yaitu seseorang yang membuat sebuah program untuk dijalankan pada komputer.
- ✓ **Sistem Analis**, yaitu orang yang biasa membuat cetak biru (*blueprint*) skema global sistem program untuk dikerjakan oleh programmer.
- ✓ **Teknisi**, yaitu orang yang bekerja untuk melakukan perawatan dan perbaikan terhadap sistem komputer.

B. Program Komputer dan Bahasa Pemrograman

Program komputer adalah kumpulan perintah atau instruksi yang disusun berdasarkan algoritma dengan menggunakan bahasa pemrograman untuk penyelesaian suatu masalah.

Bahasa pemrograman adalah perangkat lunak yang digunakan untuk menuliskan algoritma dalam bentuk teks perintah-perintah yang dapat dipahami oleh komputer. Bahasa pemrograman berfungsi memberi perintah kepada komputer untuk mengolah data sesuai dengan alur berpikir yang kita harapkan. Luaran dari bahasa pemrograman tersebut berupa program atau aplikasi, seperti program yang digunakan oleh kasir di supermarket, penggunaan lampu lalu lintas di jalan raya, dan sebagainya.

Terdapat 3 macam bahasa pemrograman yang dikenal berdasarkan hierarkinya, yaitu:

- a. Bahasa pemrograman aras dasar seperti : bahasa mesin atau bahasa *assembly*
- b. Bahasa pemrograman aras tengah seperti : bahasa C
- c. Bahasa pemrograman aras atas seperti : pascal, visual basic, MATLAB dan lain lain.

Berdasarkan definisi di atas, maka penyusunan program komputer harus memenuhi setidaknya beberapa hal berikut, yaitu:

- a. Ada masalah yang akan dipecahkan
- b. Ada algoritma untuk penyelesaian masalah
- c. Ada bahasa pemrograman untuk menterjemahkan algoritma.

C. Tahapan Menyusun Program Komputer

Langkah-langkah yang dapat dilakukan untuk menyelesaikan masalah dalam pemrograman komputer antara lain:

1. Definisi Masalah
 - ✓ Tentukan apa yang bermasalah
 - ✓ Tentukan data *input* dan *output*
2. Struktur Data
 - ✓ Tentukan variabel untuk masing-masing data
 - ✓ Tentukan tipe data untuk masing-masing variabel
3. Algoritma Program Komputer

Susunan langkah-langkah penyelesaian yang memuat: Input, Proses, dan Output.
4. Pengkodean
 - ✓ Pilih bahasa pemrograman
 - ✓ Terjemahkan algoritma ke dalam bahasa pemrograman
5. Pengujian dan Verifikasi Program
 - ✓ Uji kesalahan sintaks (penulisan program)
 - ✓ Uji kesalahan logika dengan menggunakan data input sederhana
 - ✓ Bandingkan output program dengan perhitungan manual
6. Dokumentasi program

D. Pengertian Algoritma

Algoritma berasal dari kata *Algorithm* (Inggris) yang berasal dari nama penulis buku arab yang terkenal yaitu **Abu Ja'far Muhammad Ibnu Musa Al-Khuwarizmi**. Al-Khuwarizmi dibaca orang barat menjadi *Algorism*. Al-Khuwarizmi menulis buku yang berjudul **Kitab Al Jabar Wal Muqabala** yang artinya "Buku pemugaran dan pengurangan" (*The book of restoration and reduction*). Dari judul buku itu kita juga memperoleh akar kata "Aljabar" (*Algebra*). Perubahan kata dari *algorism* menjadi *algorithm* muncul karena kata *algorism* sering dikelirukan dengan *arithmetic*, sehingga akhiran *-sm* berubah menjadi *-thm*. Kata *algorithm* inilah yang kemudian diserap dalam dalam bahasa Indonesia menjadi **algoritma**.

Algoritma adalah urutan langkah-langkah sistematis, lengkap dan logis untuk menyelesaikan suatu masalah. Kata logis merupakan kata kunci dalam algoritma. Langkah-langkah dalam algoritma harus logis dan harus dapat ditentukan bernilai salah atau benar.

Pembuatan algoritma mempunyai banyak keuntungan di antaranya:

- a. Pembuatan atau penulisan algoritma tidak tergantung pada bahasa pemrograman manapun, artinya penulisan algoritma independen dari bahasa pemrograman dan komputer yang melaksanakannya.
- b. Notasi algoritma dapat diterjemahkan ke dalam berbagai bahasa pemrograman.
- c. Apapun bahasa pemrogramannya, output yang akan dikeluarkan sama karena algoritmanya sama.

E. Syarat Algoritma atau Program Komputer yang Baik

Suatu masalah memungkinkan untuk dipecahkan dengan metode dan logika yang berlainan, sehingga suatu masalah dapat dibuatkan lebih dari satu algoritma. Untuk itu kita perlu mengetahui syarat-syarat algoritma yang baik untuk digunakan dalam bahasa program.

Beberapa persyaratan untuk menjadi algoritma atau program komputer yang baik antara lain:

1. *Realiable*, artinya hasil yang diperoleh dari proses memiliki tingkat kepercayaan yang tinggi.
2. *Efisien* (*cost* rendah), artinya proses harus diselesaikan secepat mungkin dan frekuensi kalkulasi yang sependek mungkin.
3. *Efektif*, artinya output yang dihasilkan memiliki tingkat akurasi yang tinggi (*kesalahannya* kecil).
4. *General*, artinya tidak hanya berlaku untuk menyelesaikan satu kasus saja, tapi juga untuk kasus lain yang lebih umum.
5. *Expandable* (*dapat dikembangkan*), artinya algoritma atau program dapat dikembangkan lebih jauh berdasarkan perubahan yang ada.
6. *Mudah dimengerti*, artinya siapapun yang melihat atau membacanya akan dapat memahaminya dengan jelas.
7. *Precise* (*tepat, betul, teliti*), artinya setiap perintah harus ditulis dengan seksama dan secara eksplisit tanpa mengandung unsur keragu-raguan. Setiap langkah harus jelas dan pasti. Contoh perintah yang mengandung unsur keragu-raguan adalah "*tambahkan 1 atau 2 pada x*".
8. *Terminate*, artinya algoritma harus memiliki kriteria/syarat pemberhentian.

F. Penyajian Algoritma

Algoritma dapat disajikan dengan dua teknik yaitu teknik tulisan dan teknik gambar. Algoritma yang disajikan dengan tulisan yaitu dengan struktur bahasa tertentu (misalnya bahasa Indonesia atau bahasa Inggris) dan *pseudocode*. *Pseudocode* adalah kode yang mirip dengan kode pemrograman yang sebenarnya sehingga lebih tepat digunakan untuk menggambarkan algoritma yang akan dikomunikasikan kepada pemrogram. Algoritma yang disajikan dengan gambar atau simbol-simbol (*chart*) tertentu membentuk bagan alir disebut *flowchart*.

CONTOH 1.1

Algoritma untuk menentukan upah seseorang pekerja yang dibayar berdasarkan jumlah jam kerja:

Penyajian algoritma dengan struktur bahasa

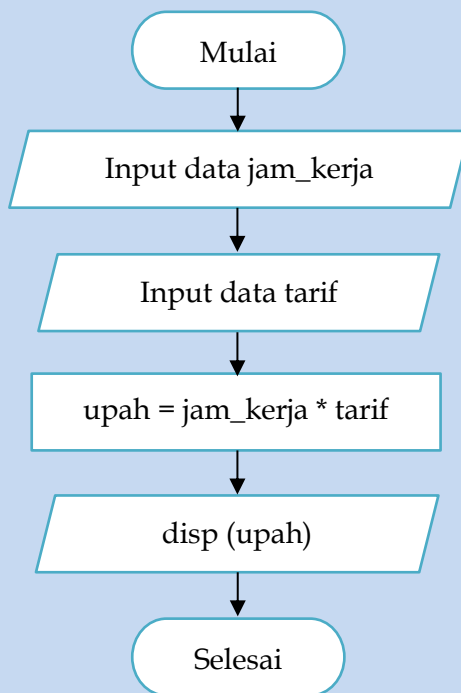
- a. Mulai
- b. Masukkan data
 - Data jam kerja
 - Data tarif per jam
- c. Hitung upah dari perkalian jam kerja dengan tarif
- d. Tampilkan gaji
- e. Akhiri

Penyajian algoritma dengan struktur *Pseudocode*

- a. Mulai
- b. Input data
 - read (jam_kerja);
 - read (tarif);

- c. Proses
upah = jam_kerja*tarif;
- d. Tampilkan output
disp (upah)
- e. Akhiri Program
end

Penyajian algoritma dengan *Flowchart*



Gambar 1.1 *Flowchart* untuk algoritma upah pekerja

1. Penyajian Algoritma dengan Tulisan

Penyajian algoritma dengan tulisan dilakukan dengan merumuskan langkah-langkah langkah -

langkah sistematis, lengkap dan logis untuk menyelesaikan suatu masalah.

CONTOH 1.2

Algoritma dengan tulisan untuk menentukan luas bangun datar persegi panjang:

Penyajian algoritma dengan struktur bahasa

2. Mulai
3. Masukkan data panjang (p) dan lebar (l)
4. Hitung luas persegi panjang, $L = p \times l$
5. Tampilkan hasil luas persegi panjang
6. Akhiri program

Penyajian algoritma dengan struktur *Pseudocode*

1. Mulai
2. Input data
read (p); read (l);
3. Proses
 $L = p * l$;
4. Tampilkan output
disp(L)
5. Akhiri Program
End

Algoritma di atas merupakan urutan logis dalam menentukan luas persegi panjang. Jika langkah ke - 3 ditukar posisinya dengan langkah ke - 4, maka algoritma yang dibuat tidak akan menemukan hasil karena salah satu variabel belum diinisialisasi nilainya. Mustahil dapat ditentukan luas persegi panjang jika salah satu dari panjang atau lebarnya belum diketahui.

CONTOH 1.3

Algoritma dengan tulisan untuk mengidentifikasi nilai dari akar-akar persamaan kuadrat:

Penyajian algoritma dengan struktur bahasa

1. Mulai
2. Tentukan nilai a, b, c
3. Hitung nilai dari $D = b^2 - 4 a c$
4. Hitung $x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$
5. Periksa apakah $D \geq 0$, jika benar, maka
 6. Tulis akar persamaan sebagai akar real
 7. Kondisi lainnya yakni ingkaran dari step 4 yaitu $D < 0$
 8. Tulis akar-akar persamaan imajiner
9. Akhiri program

Penyajian algoritma dengan struktur *Pseudocode*

1. Mulai
2. Input data
read (a); read (b); read (c);
3. Hitung nilai diskriminan
 $D = b^2 - 4 * a * c$
4. Hitung akar-akar
 $x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$
5. Proses
if $D \geq 0$,
 write (akar real)
else
 write (akar imajiner)
5. Akhiri program

Algoritma diatas menunjukkan adanya pengujian kondisi sebelum mengambil keputusan. Langkah ke-5 dan ke-6 pada struktur bahasa, tertulis sejajar dan lebih maju dari langkah ke-4. Hal ini menunjukkan bahwa langkah ke-5 dan ke-6 hanya akan dikerjakan jika kondisi pada langkah ke-4 terpenuhi. Jika kondisi step 4 tidak terpenuhi, maka pengerjaan akan berlanjut ke langkah ke-7. Algoritma jenis ini akan dibahas lebih lanjut pada bab selanjutnya.

LATIHAN 1.1

1. Jelaskan pengertian program komputer!
2. Jelaskan pengertian algoritma!
3. Jelaskan perbedaan algoritma menggunakan bahasa dan algoritma menggunakan *pseudocode* !
4. Dua buah gelas A dan B. Diketahui gelas A berisi teh dan gelas B berisi kopi. Masalahnya adalah bagaimana menukar isi gelas sehingga gelas A berisi kopi dan gelas B berisi teh? Buatlah algoritmanya.
5. Buatlah susunan algoritma untuk menentukan luas dan keliling persegi.
6. Rancang algoritma untuk menghitung luas permukaan serta voume balok yang diketahui ketiga rusuknya.
7. Sifat trikotomi bilangan real menyatakan bahwa, jika X dan Y adalah sembarang bilangan real, maka terdapat 3 kemungkinan hubungan yang terjadi antara X dan Y, yaitu $X = Y$, $X > Y$ atau $X < Y$. Buatlah algoritma yang dapat dilakukan untuk menyimpulkan hubungan X dan Y.
8. Jelaskan langkah-langkah penyusunan program computer.
9. Sebutkan syarat dan kriteria algoritma atau program komputer yang baik.

2. Penyajian Algoritma dengan *Flowchart*

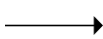
Dalam penyusunan algoritma menggunakan bahasa (Inggris/Indonesia), tahap-tahap penyelesaian masalah digambarkan dengan menggunakan kata-kata (teks). Kelemahan cara ini adalah dalam penyusunan algoritma sangat dipengaruhi oleh tata bahasa pembuatnya, sehingga terkadang orang lain sulit memahaminya. Oleh karena itu, dikembangkan metode yang menggambarkan tahap-tahap pemecahan masalah dengan merepresentasikan simbol-simbol tertentu yang mudah dimengerti dan mudah digunakan. Salah satu penulisan simbol tersebut adalah dengan menggunakan *flowchart*.



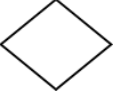



Flowchart adalah suatu diagram alur yang menggambarkan suatu algoritma program. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu sedangkan antara proses digambarkan dengan garis penghubung.

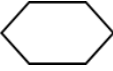



Penggunaan *flowchart* akan memudahkan kita untuk melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah. Disamping itu, *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek.

Algoritma dengan *flowchart* dilakukan dengan simbol-simbol seperti yang diperlihatkan pada Tabel 1.1 berikut.

Tabel 1.1 Simbol-simbol *flowchart* dan fungsinya

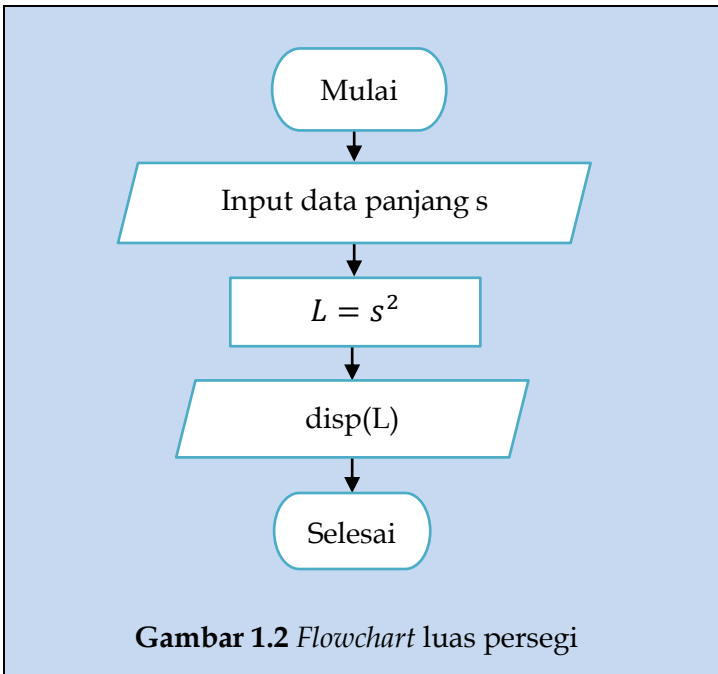
| Gambar | Nama | Keterangan |
|-------------------------------------------------------------------------------------|------------|---------------------------------------------------------------------------|
|  | Garis Alir | Menunjukkan arah aliran algoritma, dari satu proses ke proses berikutnya. |

| | | |
|-------------------------------------------------------------------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | Terminal | Menunjukkan awal atau akhir sebuah proses. |
|  | Proses / Langkah | Menyatakan kegiatan yang akan terjadi dalam diagram alir, proses perhitungan/pengolahan data. |
|  | Titik Keputusan | Proses yang menyatakan suatu pilihan berdasarkan suatu kondisi tertentu. Di titik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda. |
|  | Masukan / Luaran | Proses input/output data, parameter, informasi, inisialisasi/pemberian nilai awal. |
|  | Anotasi /Keterangan | Melambangkan komentar tentang suatu atau beberapa bagian dari diagram alir. Komentar tidak memiliki dampak apapun terhadap proses yang berlangsung. |
|  | Predefined Process | Digunakan untuk menunjukkan suatu proses yang begitu kompleks, sehingga tidak bisa dijelaskan di diagram alir ini dan merujuk pada diagram alir yang terpisah. |

| | | |
|-------------------------------------------------------------------------------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | Persiapan / Inisialisasi | Menunjukkan operasi yang tidak memiliki efek khusus selain mempersiapkan sebuah nilai untuk langkah/proses berikutnya. Lambang ini juga digunakan untuk menggantikan titik keputusan yang biasanya berbentuk ketupat jika ingin menggunakan pengulangan pada kondisi tertentu. |
|  | Konektor Dalam Halaman | Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman. |
|  | Konektor Luar Halaman | Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda |
|  | Kontrol / Inspeksi | Menunjukkan proses/langkah di mana ada inspeksi atau pengontrolan. |

Contoh 1.4

Algoritma untuk menentukan luas persegi, jika dituliskan dalam bentuk *flowchart* adalah sebagai berikut:



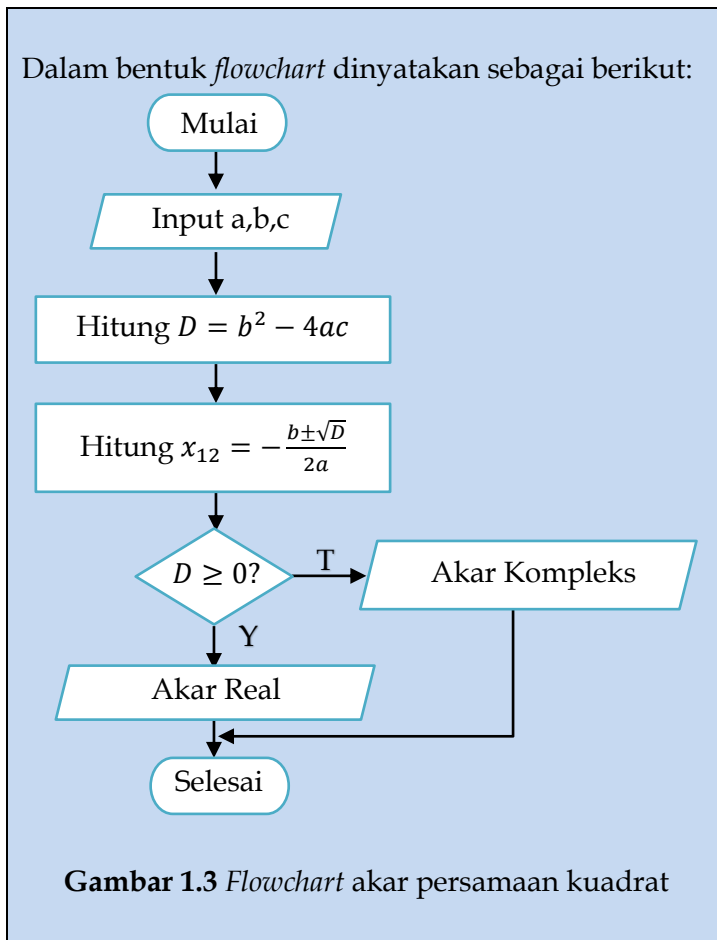
Contoh 1.5

Algoritma penentuan akar persamaan kuadrat pada Contoh 1.3, dalam bentuk *flowchart*,

Ingat kembali penyajian algoritma dengan struktur bahasa:

1. Mulai
2. Tentukan nilai a, b, c
3. Hitung nilai dari $D = b^2 - 4ac$
4. Hitung $x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$
5. Periksa apakah $D \geq 0$, jika benar, maka
 6. Tulis akar persamaan sebagai akar real
7. Kondisi lainnya ($D < 0$)
 8. Tulis akar-akar persamaan imajiner
9. Akhiri program

Dalam bentuk *flowchart* dinyatakan sebagai berikut:



Gambar 1.3 *Flowchart* akar persamaan kuadrat

LATIHAN 1.2

1. Kontruksilah algoritma bentuk *flowchart* untuk menentukan luas permukaan balok.
2. Kontruksilah algoritma bentuk *flowchart* dari algoritma trikotomi bilangan real.
3. Buatlah suatu algoritma bentuk *flowchart* untuk menghitung:
 - a) Keliling dan luas persegi panjang
 - b) Luas permukaan dan volume kubus

4. Jika diketahui Partisipasi (U_1), Nilai Tugas (U_2), UTS (U_3) dan UAS (U_4) dengan ketentuan Nilai Akhir (NA) = $0.1 \times U_1 + 0.2 \times U_2 + 0.3 \times U_3 + 0.4 \times U_4$. Buatlah algoritma Konversi Nilai dalam bentuk *flowchart* dengan catatan Nilai Akhir (NA):
- ✓ Grade NA = A jika NA ≥ 90
 - ✓ Grade NA = AB jika NA ≥ 80
 - ✓ Grade NA = B jika NA ≥ 75
 - ✓ Grade NA = BC jika NA ≥ 65
 - ✓ Grade NA = C jika NA ≥ 60
 - ✓ Grade NA = D jika NA ≥ 50
 - ✓ Grade NA = E jika NA < 50

DAFTAR REFERENSI

- Anonim. (2018, September). *Perbedaan Hardware, Software, dan Brainware*. Retrieved Januari 2019, from www.indoworx.com: <https://www.indoworx.com/perbedaan-hardware-software-dan-brainware/>
- Attaway, S. (2009). *MATLAB : A Practical Introduction to Programming and Problem Solving*. Boston: Boston University.
- Brassard, G., & Bratley, P. (1988). *Algorithm, Theory and Practice*. New Jersey: Prentice Hall.
- Brian, R., Hunt, R., Lipsman, J., & Rosenberg. (2001). *A Guide to MATLAB for Beginners and Experienced Users*. Cambridge: Cambridge University Press.
- Chabert, J. (1999). *A History of Algorithm*. Berlin: Springer-Verlag Heidelberg.
- Cormen, T., Leiserson, C., Rivest, R., & Stein, C. (2002). *Intoduction to Algorithms*. London: The MIT Press.
- Gerritsen, M. (2006). *A Brief Introduction to MATLAB*. CME200 Handout.
- Munir, R. (2007). *Algoritma dan Pemrograman, Jilid I*. Bandung: Informatika.
- Praptomo, Y. (----). *Algoritma dan Pemrograman*. Yogyakarta: STIMIK El Rahma.
- Sahyar. (2016). *Algoritma dan Pemrograman Menggunakan MATLAB*. Medan: Kencana.

- Suarga. (2004). *Algoritma Pemrograman*. Yogyakarta: Penerbit Andi.
- Suryadi, H., & Sumin, A. (1997). *Algoritma dan Pemrograman*. Jakarta: Gunadarma.
- Sutedjo, B., & Michael, A. (2000). *Algoritma dan Teknik Pemrograman*. Yogyakarta: Penerbit Andi.
- Xenophontos, C. (1999). *A Beginners Guide to MATLAB*. New York: Clarkson University.
- Yuhendra. (2013). *Algoritma dan Pemrograman dalam Bahasa Pascal*. Padang: Institut Teknologi Padang.