**Pengenalan I : Algoritma dan Peran Pemrograman dalam Algoritma**

****

Seiring perkembangan teknologi, muncul juga berbagai perangkat – perangkat yang menemani kehidupan sehari – hari seperti Smartphone, PC, Laptop dan lainnya. Perangkat – perangkat ini tidak akan dirasa berguna tanpa adanya berbagai program. Program – program yang terdapat di dalamnya diantaranya sistem operasi sebagai basis dari berjalannya program – program lain, perangkat lunak spesifik / produktifitas (seperti text processor atau spreadsheet, kalender, reminder), perangkat lunak utilitas (seperti penghemat baterai, penghapus berkas tidak berguna dll), dan bahkan game juga merupakan program untuk menemani di waktu luang atau sekedar menghabiskan waktu.

Sebagai pengguna, kita hanya tinggal menikmati keseluruhan dari apa yang ada di dalam perangkat – perangkat tersebut. Pada smartphone di zaman sekarang, kita hanya tinggal menuju store untuk mencari apa perangkat yang diinginkan kemudian melakukan instalasi. Tetapi pernahkah pengguna berpikir, bagaimana sebenarnya program – program itu dibuat ? Sebagai mahasiswa informatika, tentunya kita harus tahu lebih dalam dari pengguna awam.

Jika dilihat lebih dalam, sebuah ***program*** pada dasarnya dibuat untuk menyelesaikan ***masalah***. Kita ambil sebuah contoh misalnya sebuah program spreadsheet yang sudah sering kita gunakan yaitu Microsoft Excel. Program ini dapat menyelesaikan banyak masalah mulai dari masalah perhitungan sederhana, logika, statistika, matematik sampai pada pembuatan laporan.

Kita ambil lagi sebuah contoh program game Poker, apakah ada masalah yang diselesaikan ? Walaupun Poker cuma merupakan program Game, tetapi dibelakangnya banyak masalah – masalah yang diselesaikan agar dapat menjadi sebuah permainan yang menyenangkan untuk dimainkan. Berbagai masalah – masalah yang diselesaikan oleh poker misalnya adalah : pengacakan kartu, pembagian kartu, pengecekan kemenangan kartu, bagaimana agar AI selalu menang. Poker tentunya tidak akan menyenangkan jika ternyata pada pembagian kartu, terdapat kartu yang berulang, proses pembagian kartu yang tidak acak / berulang – ulang, dan pembayaran hadiah kemenangan yang salah). Sampai disini, kita dapat mengatakan bahwa sebuah program umumnya menyelesaikan banyak masalah sekaligus.

Pertanyaan selanjutnya adalah bagaimana program dapat menyelesaikan masalah ? Apakah ada sebuah entitas cerdas di dalamnya yang menyelesaikan masalah dan beradaptasi pada setiap permintaan – permintaan pengguna ?

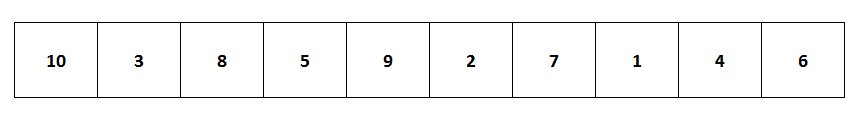
Tidak, masing - masing masalah yang diselesaikan oleh sebuah program diselesaikan oleh apa yang disebut dengan algoritma. Misal pada program Microsoft Excel, proses penggunaan rumus-rumus memiliki algoritmanya sendiri, proses membuatan laporan memiliki algoritmanya sendiri dan seterusnya. pada game Poker, proses pengacakan memiliki algoritmanya sendiri, proses pengecekan kemenangan memiliki algoritmanya sendiri dan seterusnya. Sampai disini kita tahu bahwa, sebuah program selain menyelesaikan banyak masalah juga terdiri dari banyak algoritma(jika masalah yang diselesaikan semakin banyak) dan algoritma adalah sesuatu yang digunakan untuk menyelesaikan masalah.

Untuk mengetahui lebih dalam mengenai algoritma akan dijelaskan lebih lanjut pada bab berikut Dari point ini sampai selanjutnya akan dijelaskan mengenai Masalah & Algoritma

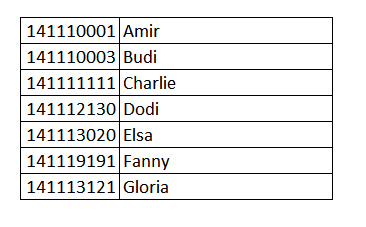
**MASALAH**

Jika sebelumnya diceritakan mengenai masalah dari sisi program, pada bab ini kita akan melihat masalah pada kehidupan sehari - hari. Coba perhatikan berbagai masalah kehidupan sehari – hari berikut ini :

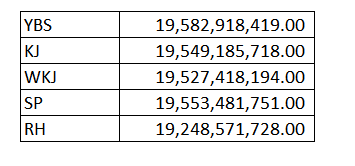
a. Anda diminta untuk menyusun voucher – voucher berikut ini menjadi terurut, sehingga pihak panitia akan mudah melakukan tracking terhadap voucher yang sudah masuk.



b. Anda diminta untuk mencari apakah NIM 141112020 merupakan peserta ujian ?

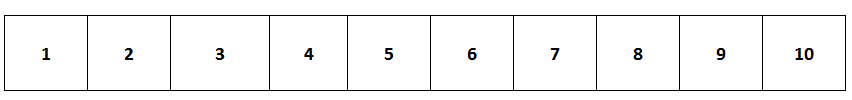


c. Anda diminta melakukan output pejabat dengan kekayaan tertinggi !

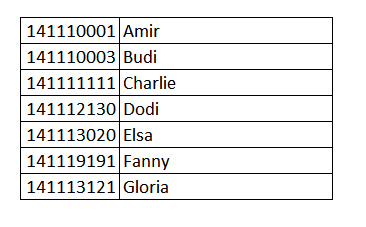


Berikut adalah solusi – solusi masalah – masalah diatas :

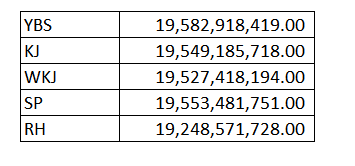
a.



b. 141112020 bukan peserta ujian



c. YBS adalah pejabat terkaya



**ALGORITMA**

Sepintas melihat pada masalah – masalah diatas, secara manusiawi kita dapat dengan mudah atau bahkan luar kepala menyelesaikan masalah – masalah diatas. Tetapi jika jumlah masalah lebih besar atau rumit, kita akan merasakan bahwa sebenarnya kita berpikir keras, kita melakukan langkah – langkah dalam menyelesaikan masalah tersebut disadari atau tidak. Pada akhirnya, kita akan menyelesaikan proses berpikir atau langkah – langkah kita kita ketika kita sudah menemukan solusi. Proses berpikir atau langkah – langkah kita dalam menyelesaikan masalah inilah yang disebut dengan ALGORITMA.

Coba bayangkan kembali, jika sebuah Rubik Cube yang belum terselesaikan adalah sebuah masalah dan Rubik Cube yang sudah selesai adalah solusinya. Maka langkah – langkah penyelesaian yang dilakukan dari masalah menjadi solusi adalah algoritmanya. Perlu diingat, algoritma yang baik adalah algoritma yang terurut dan efektif. Perhatikan kasus berikut dimana kita akan memindahkan isi dari gelas A ke gelas B dengan bantuan gelas kosong C, dapatkan Anda menentukan masalah awal, solusi, dan algoritmanya yang efektif ?

Algoritma yang baik juga harus berhenti ketika sudah menemukan solusi. Sama seperti menyelesaikan pengurutan voucher atau menyelesaikan rubik cube, kita harus berhenti jika sudah menyelesaikan masalahnya.

**PERAN PROGRAMMING PADA ALGORITMA**

Perhatikan salah satu masalah diatas yang sudah kita selesaikan sebelumnya. Anggaplah *S* merupakan himpunan masalah yang akan kita selesaikan dan *n* merupakan besar / ruang masalah yang akan diselesaikan. Dengan *S* dan *n*, masalah akan cukup mudah diselesaikan oleh manusia dan dengan cepat.

Tetapi bagaimana dengan S yang merupakan himpunan angka acak dari 1 sampai 10.000, tentunya dengan jumlah n sebesar 10.000 angka. Masalah ini juga bisa diselesaikan oleh manusia, tetapi akan membutuhkan waktu. Dengan perkembangan teknologi saat ini, masalah dengan ruang yang besar dapat diselesaikan oleh kekuatan komputasional (Komputer). Algoritma dapat diterjemahkan menjadi bahasa komputer untuk membantu menyelesaikan masalah. Algoritma dapat diterjemahkan menjadi bahasa komputer melalui pemrograman oleh seorang programmer. Disinilah peran pemrograman dan hubungannya dengan algoritma.

Terdapat banyak bahasa pemrograman dalam dunia Computer Science, kita dapat memilih salah satu yang paling kita sukai dan sesuai dengan tujuan untuk mengakomodasi kita dalam menyelesaikan masalah. Bedasarkan “kemanusiawian”, bahasa pemrograman dikategorikan menjadi dua jenis :

* **Low – Level Programming Language,** merupakan bahasa pemrograman yang lebih dekat dengan bahasa mesin. Bahasa pemrograman ini masih berupa instruksi – instruksi yang terkadang tidak memiliki arti dengan bahasa manusia normalnya.
  + Assembly, OPCODE
* **High – Level Programming Language,** merupakan bahasa – bahasa pemrograman yang dibangun diatas Low – Level Programming Language, namun memiliki bahasa yang lebih masuk akal dan lebih bisa dimengerti karena mengikuti kaidah – kaidah bahasa yang ada.
  + Pascal, Ada, Cobol, Basic, Fortran, C, C++, **Python**

Pada mata kuliah ini, kita akan menggunakan bahasa pemrogrman Python. Perlu diingat bahwa, kita bukan mempelajari bahasa pemrograman Python, namun menggunakan Python sebagai alat untuk membantu kita mempelajari penyelesaian masalah melalui pembuatan algoritma dalam bahasa pemrograman.

Tentunya sebagai bahasa pemrograman pertama yang dipelajari, kita perlu tahu bagaimana menginstalasi Python, menggunakan Python dan menjalankan algoritma pertama kita dengan menggunakan Python. Latihan pada bab ini dapat dikerjakan untuk membantu pemahaman penggunaan Python sebelum masuk ke pemahaman algoritma lebih lanjut.

**Latihan**

1. Pada latihan ini, Dosen akan mendemokan instalasi Python versi 3.3. Silahkan melakukan pencatatan dan mengajukan pertanyaan untuk mendapatkan informasi lebih lanjut mengenai instalasi Python.

2. Pada latihan ini, Dosen akan menjelaskan dan mendemokan konsep IDE pada pemrograman. Silahkan melakukan pencatatan dan mengajukan pertanyaan untuk mendapatkan informasi lebih lanjut mengenai penggunaan IDE Python.

3. Pada latihan ini, Dosen akan mendemokan beberapa algoritma sederhana yang ditulis dalam bahasa Python.

4. Pada latihan ini, Dosen akan menjelaskan dan mendemokan konsep Compiling dan Running bahasa pemrograman Python. Silahkan melakukan pencatatan dan mengajukan pertanyaan untuk mendapatkan informasi lebih lanjut mengenai penggunaan IDE Python.

**Pengenalan IIa : Struktur Dasar Algoritma**

Pada bab sebelumnya, kita mengetahui bahwa Program merupakan kumpulan dari algoritma – algoritma. Algoritma sendiri di dalamnya juga merupakan kumpulan dari struktur – struktur dasar pembangun algoritma. Terdapat tiga jenis utama struktur algoritma yaitu :

* + Sequence (Runtunan)
  + Selection / Branching (Pemilihan / Percabangan)
  + Repetition / Looping (Pengulangan)

**Sequence (Runtunan)** merupakan struktur algoritma paling dasar yang berisi rangkaian instruksi yang diproses secara sekuensial, satu per satu, mulai dari instruksi pertama sampai instruksi terakhir. Salah satu contoh Sequence(runtunan) ada pada contoh bab sebelumnya yaitu algoritma memindahkan isi dari kedua gelas.

**Selection / Branching (Pemilihan / Percabangan)** merupakan struktur algoritma yang akan digunakan dimana jika terdapat alternatif/pilihan beberapa sequence(runtunan) yang akan dijalankan jika memenuhi syarat tertentu. Percabangan dapat diibaratkan sebagai persimpangan jalan yang harus dipilih, jika kita memilih sebuah jalan dari persimpangan tersebut sudah pasti kita tidak akan menjalani yang lainnya. Salah satu contoh percabangan dapat dilihat pada contoh “Algoritma Memilih Tempat Pembuangan” berikut ini :



**Repetition / Looping (Pengulangan)** merupakan struktur algoritma yang akan digunakan dimana jika terdapat pengulangan terhadap satu pernyataan atau sequence (runtunan) tertentu. Beberapa contoh nyata penggunaan pengulangan misalnya pada saat harus melakukan penelusuran angka – angka voucher dari awal sampai akhir. Contoh lain yang lebih dekat dengan dunia nyata misalnya adalah algoritma “pembagian permen” berikut ini :

****

Percabangan dan Perulangan akan dibahas lebih lanjut pada pertemuan 3 ke atas. Perlu diperhatikan bahwa masing – masing dari struktur dasar ini tidak berdiri sendiri dan terpisah. Pada kasus pembuatan algoritma sebenarnya, struktur dasar ini akan saling dipasangkan satu sama lain. Beberapa kasus dari penggabungan struktur dasar ini adalah sebagai berikut :

* + Perulangan bersarang
  + Percabangan bersarang
  + Percabangan dalam Percabangan
  + Repetition dalam Percabangan
  + Percabangan dalam Perulangan
  + Perulangan dalam Selection
  + Perulangan dalam Perulangan dengan Selection
  + Dan kombinasi lainnya

**Pengenalan IIb: Variabel, Tipe Data & Operator**

**Pengenalan IIc : Input Proses Output**