



Evolusi Teknologi Pengaturcaraan Komputer

Pengaturcaraan komputer bermaksud menulis atur cara komputer menggunakan bahasa pengaturcaraan. Tujuan pengaturcaraan adalah untuk menghasilkan atur cara yang boleh dilaksanakan oleh komputer bagi melakukan sesuatu tugas dan menyelesaikan masalah melibatkan pengkomputan atau automasi. Secara umumnya, atur cara juga disebut sebagai program atau perisian.

Pada awal penciptaan komputer, arahan operasi komputer diberikan oleh operator komputer menggunakan input suis. Penggunaan teknologi kad tebuk kemudiannya membolehkan

atur cara direkodkan pada kad dan seterusnya komputer memproses semua arahan mengikut atur cara pada kad tersebut.

Dengan kemajuan teknologi ingatan utama dan storan, kini atur cara boleh disimpan dalam storan sekunder dalam bentuk fail binari atau fail boleh laku. Hal ini membolehkan salinan atur cara diambil terus daripada storan dan dimuatkan dalam ingatan utama jika diperlukan untuk pelaksanaan.

Secara asasnya, atur cara komputer mengandungi senarai arahan dan data yang diperlukan untuk menjalankan sesuatu tugas. Apabila

atur cara perlu dilaksanakan, atur cara akan dimuatkan dalam ruang ingatan utama yang dipanggil ingatan capaian rawak (RAM).

Selepas itu, arahan daripada atur cara itu akan dilaksanakan satu demi satu oleh pemproses sehingga pelaksanaan atur cara tamat atau ditamatkan. Semasa pelaksanaannya, atur cara mungkin turut melibatkan operasi menggunakan peranti input dan output, pemindahan data melibatkan storan sekunder atau komunikasi melalui sambungan rangkaian.

Sistem pengoperasian memerlukan lebih maklumat untuk menguruskan

pelaksanaan atur cara. Oleh sebab itu, fail boleh laku yang mengandungi atur cara disimpan mengikut format yang ditetapkan oleh sistem pengoperasian. Selain arahan atur cara, tujuan format ini digunakan adalah untuk menyusun maklumat lain yang diperlukan oleh sistem pengoperasian untuk mempersiapkan atur cara ketika hendak dilaksanakan.

Bagi sistem pengoperasian Windows misalnya, fail boleh laku menggunakan format Portable Executable (PE). Bagi sistem Linux pula, fail boleh laku disimpan dalam format Executable Linux Format (ELF), manakala MacOS menggunakan format Mach-O.

Format fail seperti PE, ELF dan Mach-O adalah untuk menyusun maklumat dalam fail boleh laku untuk kegunaan sistem. Bagi panduan pengguna, fail boleh laku boleh dikenal pasti berdasarkan sambungan nama fail. Misalnya, Windows menggunakan sambungan nama *.EXE yang bermaksud boleh laku.

Laman web FileInfo.com telah menyenaraikan sambungan bagi format fail boleh laku untuk pelbagai jenis sistem pengoperasian. Sambungan nama fail adalah berguna kepada pengguna dan aplikasi tertentu dalam membezakan antara fail boleh laku dengan fail binari yang hanya mengandungi data seperti gambar dan muzik. Walau bagaimanapun, sambungan nama fail hanyalah ciri bantuan dan boleh diubah tanpa mengikut format kandungannya.

Dari perspektif sistem komputer, atur cara atau perisian boleh dibahagikan kepada tiga kategori am, iaitu perisian terbenam, perisian sistem dan perisian utiliti atau aplikasi.

Perisian terbenam biasanya bersaiz kecil dan melaksanakan fungsi yang spesifik. Hal ini dikatakan demikian kerana perisian terbenam perlu disimpan dalam ingatan baca sahaja (ROM) yang mempunyai kapasiti storan yang kecil seperti perisian pengendali jam digital dan kalkulator. Lazimnya, pengatur cara perisian terbenam perlu mempunyai

≡ Dari sudut evolusi penciptaan, bahasa untuk pengaturcaraan boleh dikategorikan kepada bahasa mesin, bahasa himpunan dan bahasa aras tinggi. Bahasa aras tinggi merangkumi bahasa generasi ketiga dan generasi keempat.

kemahiran teknikal berkaitan dengan sistem digital dan elektronik, selain pengetahuan tentang bahasa pengaturcaraan.

Perisian sistem berperanan mengendalikan operasi berkaitan dengan platform dan fungsi sistem atau menguruskan sumber dan perkakasan termasuklah komponen sistem pengoperasian seperti kernel, antara muka pengguna, pemacu peranti, pangkalan data atau pelayan web. Pembina perisian sistem biasanya mempunyai kemahiran teknikal berkaitan dengan sistem dan platform seperti pengetahuan tentang sistem pengoperasian dan komponennya, struktur data yang digunakan serta protokol dan representasi data.

Perisian utiliti atau aplikasi pula merujuk perisian yang digunakan untuk memenuhi keperluan pengguna (atau pengguna akhir) seperti perisian permainan komputer, pemproses perkataan, penyuntingan grafik, dan pelayan Internet. Kebanyakan pengatur cara lebih cenderung membina perisian kategori yang ketiga. Hal ini mungkin disebabkan oleh faktor permintaan terhadap perisian pengguna yang lebih terbuka berbanding dengan perisian terbenam dan perisian sistem atau tahap pengetahuan teknikal yang tinggi diperlukan bagi kedua-dua kategori tersebut.

Walaupun bagaimanapun, pembina perisian pengguna juga perlu mempunyai kemahiran teknikal yang tinggi, iaitu memahami dan mentafsir keperluan dan kehendak pengguna selain berkeupayaan menggunakan bahasa pengaturcaraan dan platform untuk pelaksanaan aplikasi.

Pelbagai bahasa pengaturcaraan telah dicipta yang memperkenalkan pelbagai teknik bagi memudahkan tugas membina atur cara. Hingga 16 April 2022, laman web Wikipedia telah menyenaraikan lebih daripada 700 bahasa pengaturcaraan. Sebahagiannya adalah sangat popular, manakala sebahagian yang lain mungkin hanya pada peringkat eksperimen atau sudah tidak digunakan lagi.

Dari sudut evolusi penciptaan, bahasa untuk pengaturcaraan boleh dikategorikan kepada bahasa mesin, bahasa himpunan dan bahasa aras tinggi. Bahasa aras tinggi merangkumi bahasa generasi ketiga dan generasi keempat.

Atur cara pada awalnya ditulis dalam bahasa mesin dan boleh terus dilaksanakan oleh komputer. Namun begitu, bahasa mesin yang ditulis hendaklah menggunakan arahan berbentuk nombor keenambelasan (asas perenambelasan) yang sukar digunakan untuk membina atur cara, kecuali bagi pengatur cara yang arif tentang seni bina komputer.

Oleh sebab itu, diwujudkan pula bahasa himpunan. Arahan dalam bahasa himpunan adalah ringkas tetapi mnemonik yang bermaksud nama arahan yang digunakan menggambarkan fungsinya seperti *ADD* bagi operasi tambah atau *MOV(E)* bagi operasi pindah. Hal ini menjadikan bahasa himpunan lebih mudah digunakan berbanding dengan bahasa mesin.

Namun demikian, bahasa himpunan perlu ditukarkan kepada bahasa mesin menggunakan perisian penghimpun dan menghasilkan fail boleh laku. Bahasa himpunan biasanya

```
MOV DL, 'A' ; pindah nilai 'A' ke alat daftar DL
MOV AH, 2 ; pindah kod 2 ke alat daftar AH
INT 21H ; cetak nilai 'A'
```

Rajah 1 Sebahagian kod atur cara menggunakan bahasa himpunan. Teks selepas aksara ; ialah komen yang ditulis sebagai penerangan arahan.

digunakan untuk membina atur cara yang pendek, pustaka subrutin atau atur cara yang perlu pantas dilaksanakan. Disebabkan arahan bahasa himpunan adalah terlalu ringkas, maka bahasa himpunan sukar digunakan bagi membina penyelesaian yang kompleks.

Sebagai alternatif kepada hal tersebut, maka diwujudkan bahasa aras tinggi (bahasa generasi ketiga) yang menggunakan arahan dan struktur atur cara yang lebih dekat kepada penyelesaian masalah seperti arahan *while()* untuk ulangan atau *if-else* untuk arahan bersyarat. Selain struktur bahasa yang umum, bahasa aras tinggi juga tidak terikat kepada reka bentuk mesin (pemproses).

Hal ini adalah berbeza dengan bahasa mesin dan bahasa himpunan yang dibina mengikut reka bentuk dan sokongan pemproses. Sebagai contohnya, arahan bahasa mesin dan bahasa himpunan pemproses 64-bit seperti Intel i5 tidak sama dengan arahan untuk pemproses Intel 8080 atau 80286 walaupun semuanya dikeluarkan oleh syarikat Intel. Walau bagaimanapun, satu atur cara bahasa aras tinggi yang sama boleh dibina dan

digunakan untuk semua pemproses tersebut.

Atur cara yang ditulis dalam bahasa pengaturcaraan aras tinggi juga sering disebut sebagai kod sumber. Kod sumber perlu ditukarkan kepada bahasa mesin supaya boleh dilaksanakan oleh komputer. Proses penukaran kod sumber kepada bahasa mesin boleh dibuat menggunakan perisian pengkompil.

Pengkompil akan menghasilkan fail atur cara bahasa mesin yang boleh dilaksanakan oleh komputer tanpa memerlukan fail kod sumber. Antara bahasa pengaturcaraan yang dikompil termasuklah C, C++, Java, Visual Basic .NET, dan C# .NET. Selain Java dan .NET yang berkonsepkan mesin maya, semua pengkompil bahasa aras tinggi disediakan dalam beberapa versi mengikut mesin (pemproses) dan sistem pengoperasian tertentu supaya atur cara yang sama boleh dikompilkan kepada pelbagai versi.

Terdapat pengkompil yang membolehkan kod sumber dilaksanakan walaupun masih dalam bahasa aras tinggi. Hal ini boleh dilakukan kerana apabila kod sumber dilarikan (*run*), pengkompil

akan terus menukarkan kod sumber tersebut kepada bahasa mesin dan terus dilaksanakan kod itu. Pengkompil begini dinamakan sebagai penterjemah.

Pada kebiasaannya, hal seumpama ini melibatkan bahasa pengaturcaraan berbentuk skrip atau kod sumber terbuka seperti PHP, Perl, Python, dan Javascript. Bahasa begini direka untuk pelbagai kegunaan, memudahkan penyelesaian bagi masalah kompleks, tidak mementingkan kelajuan atau prestasi pelaksanaan serta membolehkan atur cara digunakan untuk pelbagai platform.

Bahasa aras tinggi generasi ketiga bersifat pelbagai guna dan boleh digunakan untuk menghasilkan pelbagai aplikasi. Namun demikian, terdapat bahasa pengaturcaraan yang diwujudkan pada aras lebih tinggi yang dinamakan bahasa generasi keempat. Bahasa pengaturcaraan ini digunakan untuk kegunaan sistem yang khusus. Sebagai contohnya, bahasa pertanyaan berstruktur (SQL) yang direka untuk memudahkan operasi melibatkan pangkalan data atau skrip cangkerang yang digunakan untuk melaksanakan pemprosesan berkelompok.

Pada hari ini, kebanyakan pembangunan aplikasi pengguna lebih tertumpu kepada penggunaan bahasa skrip kerana teknologi pemprosesan telah mengatasi masalah pelaksanaan bahasa skrip yang perlahan. Bahasa skrip seperti Python banyak digunakan dalam pembangunan aplikasi merentas platform seperti aplikasi web, Internet benda, kecerdasan buatan,

```
public static void main( String[] args ) {
    System.out.println("Hello Java");
}
```

Rajah 2 Sebahagian atur cara bahasa aras tinggi Java. Bahasa aras tinggi tidak terikat secara langsung kepada mana-mana reka bentuk pemproses.



dan sistem terbenam. Bahasa Python secara amnya tidak bergantung pada sistem tertentu, mempunyai pustaka subrutin yang komprehensif, lebih mudah dipelajari, pengkompil dengan kod sumber terbuka, dan mendapat sokongan yang meluas.

Semua bahasa pengaturcaraan aras tinggi mempunyai peraturan bahasa yang dipanggil sintaks. Penulisan atur cara bahasa aras tinggi perlu mengikut sintaksnya. Namun demikian, untuk menjadi pengatur cara yang mahir, adalah tidak cukup dengan sekadar mempelajari arahan dan sintaks bahasa pengaturcaraan tersebut.

Bahasa pengaturcaraan hanyalah alat untuk mengimplementasi penyelesaian. Oleh sebab itu, menguasai kemahiran bahasa pengaturcaraan sahaja belum tentu mampu membina atur cara yang bagus. Apatah lagi dengan penggunaan bahasa pengaturcaraan berasaskan skrip yang lebih mudah dipelajari.

Apabila menghasilkan atur cara, perkara utama yang perlu ada bagi seorang pengatur cara ialah kemahiran merangka langkah penyelesaian untuk menghasilkan

output yang diharapkan. Pada kebiasaannya, langkah penyelesaian suatu masalah boleh dibuat menggunakan lebih daripada satu cara.

Oleh itu, pengatur cara mestilah mampu memilih langkah penyelesaian yang terbaik daripada pilihan penyelesaian yang ada. Menguasai kemahiran ini juga memerlukan masa, banyak latihan dan pengalaman.

Selain itu, pengatur cara juga perlu mahir dengan konsep penyelesaian seperti pengaturcaraan berstruktur, modular atau berorientasikan objek. Membina atur cara bukan sahaja melibatkan menulis atur cara tetapi juga memastikan kod atur cara itu dapat disemak semula untuk dibaiki dan diselenggara jika perlu.

Ada masanya pembina atur cara tidak dapat menyanggah kod atur caranya lagi dan diambil alih oleh orang lain. Oleh sebab itu, binaan atur cara yang mengikut standard penulisan dapat memudahkan penyenggaraan. Penggunaan komen sebagai dokumentasi dalam kod atur cara juga harus diamalkan.

Disebabkan keperluan pembangunan pada masa ini yang semakin kompleks dan kemunculan

pelbagai teknologi baharu, pengatur cara juga perlu mahir menggunakan alat sokongan pembangunan. Antara alat yang berguna termasuklah persekitaran pembangunan bersepadu (IDE) seperti Eclipse; Visual Studio Code dsb.; penyahpepijat dan alat pemeriksa; sistem pengurus versi seperti Git dan CVS; dan menggunakan pustaka subrutin.

Kesemua program ini lazimnya menawarkan fungsi yang memudahkan tugas membina atur cara yang menjimatkan masa, tenaga dan kos. Pengatur cara juga perlu mempunyai pengetahuan mengenai platform aplikasi yang dibangunkan seperti aplikasi meja (*desktop*), web, mobil, dan rangkaian awan.

Sebagai kesimpulannya, atur cara komputer dibina untuk membolehkan komputer membantu pengguna menyelesaikan tugas pengkomputan dan automasi. Selain memahami teknologi dan evolusi berkaitan dengan pembinaan atur cara, pengatur cara perlu mempunyai kemahiran membina atur cara sebagai langkah untuk menyelesaikan masalah. Pengatur cara juga perlu memastikan atur cara dapat menghasilkan output yang betul, berstruktur dan mudah diselenggara. ⁴⁸