

Spesifikasi Bahasa Pengaturcaraan AMAL

oleh
Safie Bin Mat Yatim
Noriah bte Idris
Mohd Seth Bin Yaacob

Fakulti Sains Komputer dan Sistem Maklumat
Universiti Teknologi Malaysia,
Jalan Semarak, Kuala Lumpur.

Abstrak

Bahasa pengaturcaraan telah menjadi alat komunikasi yang penting di antara manusia dan komputer. Generasi bahasa komputer telah mengalami evolusi, yang bermula dari generasi pertama berbentuk bahasa mesin, menuju ke generasi keenam yang menghampiri bahasa tabii. Kertas ini akan mengutarakan spesifikasi satu bahasa pengaturcaraan baru yang dinamakan AMAL. Bahasa Malaysia dijadikan asas bagi binaan-binaan bahasa tersebut dan direkabentuk berdasarkan ciri-ciri yang terdapat di dalam bahasa pengaturcaraan Pascal, Ada dan C.

Katakunci: Bahasa Pengaturcaraan, Spesifikasi Bahasa, AMAL.

Abstract

Programming languages have become an important tool for communication between human and computers. The generation of computer languages has evolved, starting from the first generation in the form of machine language to the sixth generation, which is prone to natural language. This paper will highlight a specification of a new programming language, called AMAL. The language constructs of this programming language use Bahasa Malaysia and is based on the existing programming languages Pascal, Ada and C.

1.0 Pendahuluan

Bahasa pengaturcaraan telah menjadi alat komunikasi terpenting di dalam hubungan manusia-komputer. Dengan bahasa pengaturcaraan, manusia dapat menjadikan komputer sebagai "alat bantu" yang sangat berguna di pelbagai bidang penghidupan. Tanpa bahasa pengaturcaraan, kebolehan dan penggunaan sesuatu jenis komputer tidak akan seanjal (fleksibel) seperti penggunaannya sekarang, dan setiap masalah yang perlu diselesaikan melalui komputer akan memerlukan komputer yang khusus dibina untuk mengatasi masalah tersebut. Memandangkan kepentingan ini, bahasa pengaturcaraan merupakan salah satu bidang yang sentiasa mendapat tumpuan oleh penyelidik-penyelidik sains komputer. Hasil daripada penyelidikan-penyelidikan yang berterusan selama hampir 40 tahun, bahasa pengaturcaraan telah mengalami elovusi. Elovusi tersebut bermula dengan bahasa generasi pertama yang berbentuk bahasa mesin dan merupakan bahasa pengaturcaraan peringkat yang paling rendah, menuju ke bahasa

Kertaskerja ini disunting oleh: En Aziz Romli.

generasi keenam yang mengahampiri bahasa tabii. Pada hari ini, mereka yang terlibat dengan dunia komputer, tentu telah biasa melihat, mendengar dan/atau menggunakan bahasa-bahasa pengaturcaraan yang popular seperti COBOL, FORTRAN, C dan Pascal. Bahasa-bahasa tersebut merupakan sebahagian daripada bahasa pengaturcaraan yang digolong sebagai bahasa generasi ketiga. Kertas kerja ini akan membincangkan spesifikasi bahasa pengaturcaraan AMAL, yang juga dikelaskan dalam kategori generasi ketiga. Bahasa pengaturcaraan ini berbeza daripada bahasa-bahasa pengaturcaraan yang lain dari segi binaan bahasanya yang berasaskan kepada Bahasa Malaysia.

2.0 Objektif Bahasa Pengaturcaraan

Dari perspektif sejarah, objektif bahasa pengaturcaraan yang utama adalah sebagai "alat bantu" bagi pengaturcara komputer dalam melaksanakan tugas-tugas pengaturcaraan mereka. Kebanyakan bahasa pengaturcaraan telah direkacipta untuk tujuan-tujuan utama yang tertentu, seperti :-

COBOL	- perniagaan
FORTRAN	- saintifik
PROLOG	- kecerdasan buatan
SNOBOL	- pemprosesan teks

Walaupun demikian, bahasa-bahasa pengaturcaraan popular pada hari ini, seperti Pascal, C, MODULA-2 dan ADA lebih bersifat umum, di mana ianya boleh digunakan dalam membangunkan peibagai applikasi. AMAL direkabentuk dengan objektif utamanya :-

Mewujudkan sesuatu bahasa pengaturcaraan baru yang bersifat umum, menggunakan Bahasa Malaysia sebagai asas pembentukan binaan-binaannya, serta menjadi bahasa pengaturcaraan yang boleh digunakan di peringkat Universiti, khususnya untuk tujuan pengajaran dan pembelajaran.

3.0 Evolusi Bahasa Pengaturcaraan

Seperti yang telah dijelaskan sebelum ini, rekabentuk bahasa pengaturcaraan peringkat tinggi lebih hampir kepada bahasa tabii manusia. Ini memudahkan pengguna-pengguna komputer terutamanya pengaturcara komputer menggunakannya. Berdasarkan kepada evolusi bahasa pengaturcaraan, kebanyakan bahasa-bahasa pengaturcaraan baru telah menggunakan bahasa-bahasa pengaturcaraan yang terlebih dahulu dicipta sebagai asas bagi pembentukannya. Fakta ini boleh dilihat pada rajah 1.1 yang menunjukkan perkembangan bahasa pengaturcaraan dari dekad 50an sehingga dekad 80an.

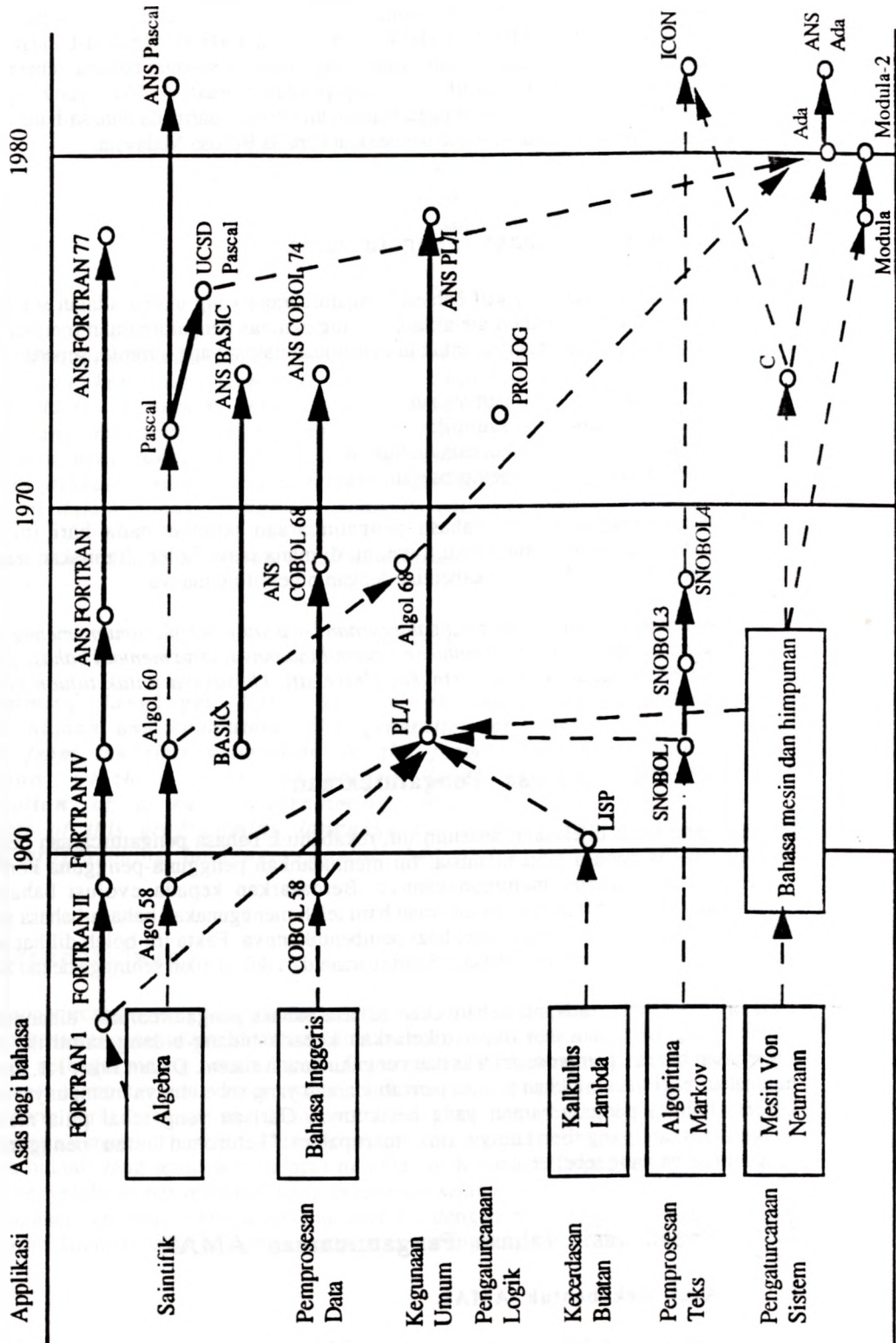
Dari rajah 1.1 juga, didapati kebanyakan bahasa-bahasa pengaturcaraan dibangunkan untuk tujuan khusus yang berlainan dan dapat dikelaskan kepada bidang-bidang saintifik, pemprosesan data, kecerdasan buatan, pemprosesan teks dan pengaturcaraan sistem. Dalam rajah 1.1, garisan "putus-putus" digunakan bagi menunjukkan bahasa pengaturcaraan yang sebelumnya mempunyai pengaruh yang kuat ke atas bahasa pengaturcaraan yang berikutnya. Garisan yang tebal pula menunjukkan bahasa pengaturcaraan yang berikutnya itu merupakan "keturunan" atau pengganti kepada bahasa pengaturcaraan yang sebelumnya.

4.0 Spesifikasi Bahasa Pengaturcaraan AMAL

4.1 Asas Rekabentuk AMAL

Dalam merekabentuk bahasa pengaturcaraan AMAL, kajian terhadap bahasa-bahasa pengaturcaraan yang sedia ada telah dijalankan. Hasil daripada kajian tersebut, beberapa bahasa pengaturcaraan telah dipilih untuk dijadikan asas bagi rekabentuk AMAL. Bahasa-bahasa pengaturcaraan yang telah dipilih adalah Pascal, C dan juga ADA. Ketiga-tiga bahasa tersebut dipilih berdasarkan kepada faktor-faktor yang berikut:-

Rajah 1.1 : Perspektif sejarah bahasa-bahasa pengaturcaraan yang terdapat pada hari ini.



- i. Kebanyakan bahasa sekarang mengamalkan konsep pengaturcaraan berstruktur, iaitu satu konsep yang menjadikan penulisan aturcara lebih tersusun dan bersistematik. Ketiga-tiga bahasa pengaturcaraan di atas mempunyai ciri-ciri ini.
- ii. Ketiga-tiga bahasa pengaturcaraan di atas mempunyai nahu yang mantap bagi pernyataan-pernyataan berkumpulan.
- iii. ADA pula mempunyai nahu yang nyata. Ini memudahkan pengguna-pengguna dan proses pengkompilan aturcaranya. Ianya juga membolehkan sebuah pengkompil yang mempunyai sistem pengendalian ralat yang lebih berkesan dapat dihasilkan.
- iv. Pascal pula adalah satu bahasa pengaturcaraan yang sangat sesuai untuk pengajaran, kerana strukturnya yang mudah dan mirip kepada Bahasa Inggeris yang selalu digunakan.
- v. C dan ADA mempunyai kebolehan pengaturcaraan peringkat rendah seperti yang terdapat di dalam bahasa himpunan.
- vi. C dan Pascal boleh dikatakan berjaya dari segi mudahalihnya (kebolehan untuk digunakan pada komputer-komputer yang berlainan). Ini adalah disebabkan kedua-duanya telah direkabentuk supaya bebas dari sebarang ciri perkakasan sesebuah komputer.

4.2 Spesifikasi Bahasa

Rekabentuk bahasa pengaturcaraan kebiasaannya melibatkan penulisan spesifikasi bagi bahasa pengaturcaraan tersebut. Di dalam spesifikasi bahasa pengaturcaraan perkara-perkara utama yang diambil kira adalah :-

- i. Jenis-jenis data
- ii. Struktur kawalan
- iii. Kawalan data dan subprogram

Perbincangan mengenai spesifikasi bahasa pengaturcaraan AMAL akan meliputi perkara-perkara berikut :-

- i. Katakunci
- ii. Pencam
- iii. Pemalar
- iv. Jenis-jenis data
- v. Struktur kawalan
- vi. Kawalan data dan subprogram

4.2.1 Katakunci

Setiap bahasa pengaturcaraan yang telah direkabentuk mempunyai senarai katakunci-katakunci mereka yang tersendiri. Penggunaan katakunci adalah berbeza di antara satu bahasa pengaturcaraan dengan bahasa pengaturcaraan yang lain; ada yang membenarkan penggunaan katakunci sebagai nama bagi pembolehubah dan yang lain sebaliknya. Bagi bahasa AMAL, katakunci adalah perkataan yang telah dikhaskan penggunaannya. Ini bermakna perkataan-perkataan yang membentuk senarai katakunci tersebut tidak boleh digunakan sebagai nama bagi pembolehubah-pembolehubah.

Bilangan katakunci merupakan salah satu perkara penting yang perlu dititikberatkan dalam merekabentuk sesuatu bahasa. Ini disebabkan, bilangan katakunci secara tidak langsung akan menggambarkan:

- i. Kemampuan sesuatu bahasa untuk mewakili penyelesaian masalah ke dalam bentuk aturcara yang akan dilaksanakan oleh komputer.
- ii. Jangka masa yang diperlukan untuk mempelajari dan memahami bahasa tersebut. Pada umumnya bahasa yang mempunyai senarai katakunci yang panjang lebih sukar untuk dipelajari berbanding dengan bahasa yang mempunyai bilangan katakunci yang lebih kecil.

Dalam merekabentuk AMAL, perhatian yang serius diberikan terhadap dua faktor di atas dalam percubaan untuk menghasilkan satu bahasa yang mempunyai bilangan katakunci yang kecil (41) tanpa mengorbankan kebolehfungsian bahasa tersebut sebagai bahasa untuk kegunaan pelbagai bidang. Senarai katakunci bagi AMAL adalah seperti berikut:

atau	fungsi	maka	rekod	ulang
aturcara	gelung	menurun	rentetan	
bagi	hingga	mula	sebaliknya	
bahagi	jenis	nil	segmen	
bukan	jlka	nilaiawal	sehingga	
cabang	ke	padat	selagi	
dalam	kes	pilih	selainnya	
dan	kesatuan	pmalar	set	
dengan	label	prosidur	tamat	
fail	lakukan	pubah	tatasusunan	

4.2.2 Pemalar

Pemalar merupakan satu nilai yang tetap dan tidak berubah. Terdapat 2 bentuk pemalar yang selalu disediakan oleh kebanyakan bahasa pengaturcaraan, iaitu : nombor dan harafiah. Di dalam bahasa pengaturcaraan AMAL, jenis data bagi pemalar yang ditakrifkan adalah sah (real), integer, harafiah dan juga rentetan. Berikut adalah contoh-contoh pemalar bagi setiap jenis data tersebut :-

sah	1.245, 3.0e3, 1e-1
integer	1, 2, 3
harafiah	'1', 'a', 'b'
rentetan	'abcdef', '123'

4.2.3 Pencam

Pencam merupakan satu siri aksara-aksara (characters), yang mana ianya boleh menjadi katakunci atau nama bagi pembolehubah. Di dalam bahasa pengaturcaraan AMAL, panjang maksima bagi pencam ialah 32 aksara. Walaupun begitu, hanya 8 aksara pertama sahaja yang akan diambil kira atau yang bererti. Aksara-aksara yang dibenarkan bagi membentuk pencam adalah :-

abjad	- huruf-huruf A hingga Z, a hingga z
nombor	- 0 hingga 9
aksara khas	- \$ dan _

Bagi aksara abjad, tidak ada perbezaan di antara huruf besar dan huruf kecil. AMAL menganggap keduanya adalah sama di antara satu sama lain. Aksara permulaan bagi pencam mesulah abjad.

Contoh-contoh pencam yang dibenarkan :-

JUMLAH GAJIS NOMBOR_1 Jumlah_Besar

Contoh-contoh pencam yang tidak dibenarkan :-

JUMLAH-BESAR - Ada aksara yang tidak dibenarkan
 NOMBOR 1 - Ada ruang kosong
 \$GAJI - Aksara permulaan bukan abjad

4.2.4 Jenis-jenis Data

Jenis-jenis data dapat dikelaskan kepada dua jenis, iaitu: skalar dan gubahan. Jenis data skalar hanya mempunyai satu elemen data sahaja di dalamnya, manakala jenis data gubahan adalah sebaliknya. Jenis-jenis data skalar bahasa pengaturcaraan AMAL adalah sama dengan jenis data skalar yang terdapat di dalam bahasa Pascal, iaitu: sahih, integer, boolean, aksara dan pengangkaan (enumerated). Rentetan tidak dikelaskan sebagai jenis skalar kerana ianya dibentuk daripada elemen-elemen data aksara yang digabungkan dalam satu tatasusunan. Berikut ini ditunjukkan bentuk-bentuk pengisytiharan di dalam bahasa AMAL bagi pembolehubah yang mempunyai jenis data skalar.

pubah		
nombor	: INTEGER ;	- Jenis integer
huruf	: AKSARA ;	- Jenis aksara
ujian	: BOOLEAN ;	- Jenis boolean
hasil	: SAHIH ;	- Jenis sahii
digit	: (SATU, DUA, TIGA) ;	- Jenis pengangkaan

Jenis-jenis data gubahan yang dilaksanakan di dalam bahasa pengaturcaraan AMAL ialah tatasusunan, rekod, penunjuk, fail, set dan kesatuan. Bentuk-bentuk struktur data ini adalah bersamaan dengan jenis data gubahan yang terdapat bahasa Pascal dan juga C. Berikut ini merupakan contoh-contoh bagi setiap jenis data gubahan. Jenis data kesatuan berbeza daripada rekod dari segi penggunaan ingatan. Medan-medan di dalam rekod mempunyai lokasi-lokasi ingatan mereka sendiri, manakala medan-medan di dalam kesatuan merujuk kepada lokasi ingatan yang sama.

a) **Tatasusunan**

Contoh :-

A : Tatasusunan (.1..10.) jenis Integer ;

b) **Rekod**

Contoh :-

rekpel = Rekod
 nometrik : Integer ;
 nama : Rentetan(40) ;
 Tamat ;

c) **Fail**

Contoh :-

failp = Fail jenis rekpel ;

d) **Penunjuk**

Contoh :-

pnod = @rekpel ;

e) **Set**

Contoh :-

Set_aks = Set jenis aksara ;

f) **Kesatuan**

Contoh :-

rekpel = Kesatuan
 nometrik : Integer ;
 nama : Rentetan(40) ;
 Tamat ;

4.2.5 Struktur Kawalan

Struktur kawalan di dalam aturcara akan mengandungi pernyataan-pernyataan yang dibentuk berdasarkan kepada ciri-ciri berikut :-

- a) Pernyataan berjjukan
- b) Pernyataan beralternatif
- c) Pernyataan berulang

Jenis-jenis pernyataan pula boleh dikelaskan kepada jenis-jenis berikut :-

- ungkapan
- pernyataan umpukan
- pernyataan bersyarat
- pernyataan pilihan
- pernyataan ulangan
- panggilan subprogram

a) Ungkapan

Ungkapan dalam bahasa AMAL dibahagikan kepada 2 kategori, iatu: ungkapan boolean dan ungkapan aritmetik. Contoh-contoh ungkapan adalah seperti berikut :-

Ungkapan boolean : $Y > Z$
 Ungkapan aritmetik : $B * A * \text{SIN}(\text{TETA}) + \text{COS}(\text{TETA})$;

b) Pernyataan umpukan

Bentuk umum pernyataan umpukan adalah :-

<pembolehbubah> := <ungkapan> ;

dengan simbol "==" menunjukkan pengumpulan nilai ungkapan kepada pembolehubah. Contoh pernyataan umpukan adalah seperti berikut :-

UJI := $B**2 - 4*A*C$;

c) Pernyataan bersyarat

Contoh pernyataan bersyarat adalah seperti berikut :-

Jika $X = Y$ maka
 $X := 0$;
 Tamat Jika ;

atau

Jika $X = Y$ maka
 $X := 0$;
 Sebaliknya
 $X := -1$;
 Tamat Jika ;

d) Pernyataan pilihan

Contoh pernyataan pilihan adalah seperti berikut :-

Pilih (X)
 Kes 1 : $Y := Y+1$;
 Kes 2 : $Y := Y+2$;
 Tamat Pilih ;

atau

Pilih(X)
 Kes 1 : $Y := Y+1$;
 Selainnya
 $Y := 0$;
 Tamat Pilih ;

- e) **Pernyataan ulangan**
Terdapat empat jenis bentuk pernyataan ulangan iaitu **BAGI**, **SELAGI**, **ULANG-SEHINGGA** dan **GELUNG**.

i) **Pernyataan BAGI**

Contoh pernyataan :-

```
Bagi I := 1 hingga 10 lakukan
    Jumlah := Jumlah + 1 ;
Tamat Bagi ;
```

ii) **Pernyataan SELAGI**

Contoh pernyataan :-

```
Selagi X = Y lakukan
    X := X - 1 ;
Tamat Selagi ;
```

iii) **Pernyataan ULANG-SEHINGGA**

Contoh pernyataan :-

```
Ulang
    X := X + 1 ;
Sehingga X = Y ;
```

iv) **Pernyataan GELUNG**

Contoh pernyataan :-

```
Gelung
    X := X+1 ;
    Jika X = Y maka
        Keluar ;
    Tamat Jika ;
Tamat Gelung ;
```

f) **Panggilan subprogram**

Pernyataan panggilan subprogram adalah terdiri daripada :-

- a) Panggilan tatacara
- b) Panggilan fungsi di dalam ungkapan

Contoh panggilan tatacara :-

```
KIRA(X, Y)
```

di mana KIRA adalah suatu tatacara (procedure).

Contoh panggilan fungsi di dalam ungkapan :-

```
X = JUMLAH(Y) ;
```

di mana JUMLAH adalah suatu subprogram berbentuk fungsi.

4.2.6 Kawalan data dan Subprogram

Apa yang dimaksudkan dengan kawalan data adalah aliran dan skop data di dalam satu-satu aturcara. Struktur bahasa pengaturcaraan AMAL adalah berbentuk blok seperti bahasa pengaturcaraan Pascal. Bentuk struktur blok ini menyebabkan data-data atau pembolehubah-pembolehubah di dalam aturcara mempunyai skop mereka yang unik. Skop data atau pembolehubah ini boleh sama ada berbentuk sejagat (global), bukan setempat (non-local) ataupun setempat (local). Hubungan di antara subprogram melalui parameter boleh dilalukan melalui penghantaran parameter secara rujukan dengan alamat (call by reference) ataupun melalui rujukan dengan nilai. Hayat dan skop bagi data adalah sama seperti yang terdapat di dalam bahasa pengaturcaraan Pascal.

4.2.6.1 Subprogram

Subprogram di dalam bahasa pengaturcaraan AMAL akan menggunakan katakunci PROSIDUR bagi menandakan **tatacara** dan FUNGSI bagi menandakan **fungsi**.

Contoh-contoh subprogram :-

Tatacara

Prosidur Kira (x : Integer) ;

Mula

x := x+1 ;

Tamat Kira ;

Fungsi

Fungsi Jumlah (Y : snom) : Integer ;

Pubah

s, i : Integer ;

Mula

s := 0 ;

Bagi i := 1 **hingga** 10 **lakukan**

s := s+Y(i.) ;

Tamat Bagi ;

Jumlah := s ;

Tamat Jumlah ;

4.2.6.2 Skop data di dalam subprogram

Perkongsian data di antara subprogram-subprogram dapat dilaksanakan melalui dua cara, iaitu :-

- a) Penghantaran parameter.
- b) Penggunaan data-data 'global' atau 'non-local'.

Segmen aturcara berikut menunjukkan skop-skop pembolehubah seperti yang dimaksudkan di atas :-

Aturcara Pertama ;

Pubah

x : Integer ;

Prosidur Kira ;

Pubah

y : Integer ;

Prosidur Ganda ;

Pubah

I : Integer ;

Mula

I := 2 ;

x := y**I ;

Tamat Ganda ;

Mula (* Kira *)

 y := 1;

 Ganda ;

Tamat Kira ;

Mula (* Petama *)

 Kira ;

 Tulis (x) ;

Tamat Pertama ;

Dari aturcara di atas :-

- X adalah pembolehubah 'global'
- Y adalah pembolehubah 'non-local'
- I adalah pembolehubah 'local'.

5.0. Penutup

Bahasa pengaturcaraan AMAL dengan spesifikasi bahasa yang dijelaskan di atas telah dilaksanakan di dalam dua versi. Versi pertama telah dilaksanakan dengan membina sebuah pengkompil pada komputer kerangka utama IBM. Versi kedua pula dilaksanakan pada komputer mikro (IBM-PC) sebagai sebuah pentafsir yang melaksanakan kod mesin maya yang dikenali sebagai mesin-M. Di dalam perlaksanaannya, pengkompil dan pentafsir AMAL juga dilengkapi dengan beberapa kemudahan untuk pengguna menyediakan aturcara yang dapat memenuhi keperluan penulisan aturcara yang baik. Salah satu kemudahan tersebut adalah "pencetak cantik" yang membolehkan aturcara disusun mengikut struktur atau format yang seragam.

Secara keseluruhannya bahasa AMAL merupakan satu bahasa yang dibangunkan berasaskan bahasa Pascal dengan tambahan ciri-ciri tertentu yang diambil daripada bahasa C dan Ada. Di samping itu, AMAL berbeza daripada bahasa Pascal terutamanya dari segi penggunaan katakunci dan pembentukan binaan-binaan aturcaranya yang berasaskan Bahasa Malaysia, berbanding dengan bahasa Pascal yang menggunakan bahasa Inggeris. ■

Rujukan dan Bibliografi

Tucker, Allen B., 1985. Programming Languages, McGraw-Hill, 1985.

Pratt, Terence W., 1984. Programming Languages: Design and Implementation, Prentice-Hall, 1984.

Tremblay, Jean-Paul and Sorenson, Paul G., 1985. The Theory and Practice of Compiler Writing, McGraw-Hill, 1985 (pages 68-134).

Ramlan Mahmood, Md Yazid Mohd Saman, Mohd Sidique Abdul Hamid, 1989. AWAL Interpreter - The Specification of the Language - Report No. 35/89, Computer Science Unit, Mathematics Department, Universiti Pertanian Malaysia, 1989.