

UNIVERSITI SAINS MALAYSIA
Peperiksaan Semester Tambahan
Sidang 1986/1987

CSP301 - Bahasa Programan

Tarikh: 23 Jun 1987

Masa: 9.00 pagi - 12.00 tgh.
(3 Jam)

Kertas ini terdiri daripada dua bahagian - bahagian A dan bahagian B.

Bahagian A terdiri daripada satu (1) soalan dan soalan ini wajib dijawab.

Bahagian B terdiri daripada empat (4) soalan. Anda diperlukan menjawab mana-mana tiga (3) daripada empat soalan itu.

Soalan 1 membawa markah 40%, manakala soalan-soalan lain membawa 20% setiap satu.

Semua soalan mesti dijawab dalam Bahasa Malaysia.

BAHAGIAN A

1. (a) (i) Apakah perbezaan di antara pasangan-pasangan sebutan di bawah? Bagi setiap kes, berikan contoh(-contoh) yang sesuai untuk menjelaskan jawapan anda.

1. 'ungkapan' dan 'pernyataan'
2. 'skop' dan 'hayat' pembolehubah

(ii) Terangkan maksud setiap sebutan di bawah. Bagi setiap kes, berikan contoh(-contoh) yang sesuai untuk menjelaskan jawapan anda.

1. tatasusunan dinamik
2. semakan jenis
3. kesan sampingan

(50/200)

- (b) (i) Jelaskan bagaimana suatu 'ungkapan Polish' boleh diterbitkan daripada suatu ungkapan infix aritmetik.
- (ii) Terangkan kaedah untuk menilaikan suatu ungkapan Polish.
- (iii) Jelaskan mengapa setiap simbol bukan terminal S (kecuali simbol mula) yang terdapat dalam sesuatu set peraturan Backus-Normal harus memenuhi dua syarat yang berikut:

1. $M \Rightarrow^* xSy$

2. $S \Rightarrow^* r$

di mana

M ialah simbol mula

x dan y adalah sebarang rentetan

r ialah rentetan yang tidak mengandungi simbol bukan terminal

(50/200)

- (c) (i) Dalam tafsiran bertatacara, proses penyatuan boleh dikatakan memainkan peranan 'menyampaikan parameter'? Jelaskan.

- (ii) Leraikan dua klausa berikut:

$$P(f(x),x,A) \leftarrow Q(x,z), R(z,y)$$

$$T(x,y,y) \leftarrow M((x,y), P(f(B),y,x)$$

(50/200)

- (d) (i) Terangkan bagaimana senarai sekutu ('association list') dan senarai sifat ('property list') diguna dalam sistem LISP untuk menyimpan nilai atom.

- (ii) Terangkan peranan dan fungsi kedua-dua bentuk LISP di bawah:

1. fungsi QUOTE

2. ungkapan LAMBDA

(50/200)

BAHAGIAN B

2. (a) Apakah yang dimaksudkan apabila kita kata sesuatu bahasa programan itu menguatkuasakan penjenisan kuat? Apakah faedah penjenisan kuat?

(30/100)

- (b) Perkembangan bahasa-bahasa programan, i.i. penambahan/penyingkiran konsep-konsep atau ciri-ciri bahasa, adalah setimpal dengan objektif kejuruteraan sofwer: "untuk menyokong pembangunan sofwer yang betul, boleh dipercayai dan mudah diselenggara". Berikut adalah beberapa ciri bahasa programan. Bagi setiap satu, bincangkan kesannya terhadap objektif kejuruteraan sofwer itu.

(i) pernyataan GOTO

(ii) struktur blok

(iii) jenis data takrif-pengguna (i.i. yang ditakrifkan oleh juruprogram, seperti yang boleh dibuat dalam PASCAL.)

(30/100)

- (c) Andaikan anda dipetugaskan untuk merumuskan suatu bahasa programan baru dan bahasa itu harus menyokong rutin. Bincangkan unsur-unsur (ciri bahasa) yang akan anda sediakan untuk mengisytihar dan menggunakan rutin. Bagi setiap unsur yang diutarakan, jelaskan mengapa ia difikir penting atau berguna. Peningkan semantik (i.i. maksud sesuatu ungkapan) daripada sintaksis dalam jawapan anda.

(40/100)

3. (a) Suatu peraturan Backus-Normal yang melibatkan rekursi kiri ditunjuk di bawah:

$$U ::= xa \mid yp \mid Uc \mid xb \mid Ud \mid yq$$

di mana x,y,a,b,c,d,p dan q adalah simbol-simbol terminal.

Dengan merujuk kepada peraturan di atas sebagai contoh, terangkan dan tunjukkan bagaimana rekursi kiri boleh digantikan dengan satu peraturan yang setara yang tidak melibatkan rekursi kiri.

(30/100)

- (b) Jelaskan bahagian-bahagian utama yang terdapat dalam suatu pengkompil serta peranan/fungsi setiap bahagian.

(40/100)

- (c) Berikut ialah suatu set peraturan-peraturan Backus-Normal yang menspesifikasikan sintaksis pernyataan umpukan untuk sesuatu bahasa khayalan:

```
<ayat umpukan> ::= <p_ubah> := <ungkapan aritmetik>
<ungkapan aritmetik> ::=
    <sebutan> | <ungkapan aritmetik> + <sebutan> |
    <ungkapan aritmetik> - <sebutan>
<sebutan> ::= <faktor> | <sebutan> * <faktor> | <sebutan> / <faktor>
<faktor> ::= <primer> | <faktor> ^ <primer>
<primer> ::= <p_ubah> | <nombor> | (<ungkapan aritmetik>)
<(p_ubah> ::= <pencam> | <pencam> (<senarai subskrip>)
<senarai subskrip> ::= <ungkapan aritmetik> |
    <senarai subskrip>, <ungkapan aritmetik>
```

Dengan merujuk kepada set peraturan ini, lukiskan pohon penghuraian ("parse tree") untuk pernyataan umpukan berikut:

$x[6] := a + b \uparrow (3 * c) / 22$

(30/100)

4. Peringatan: Dalam pernyataan-pernyataan logik dalam soalan ini, pencam yang bermula dengan huruf kecil adalah simbol pembolehubah manakala pencam yang bermula dengan huruf besar adalah simbol konstan.

- (a) Jelaskan, melalui contoh, maksud taakulan atas-ke-bawah dan terangkan mengapa ia dikatakan sebagai "menuju matlamat".

(25/100)

- (b) Predikat "sambung (x,y,z)" benar jika senarai z adalah sama dengan senarai yang didapati jika senarai x disambung dengan senarai y. Predikat ini ditakrifkan seperti berikut:

```
sambung([],y,y).  
sambung([u|x],y,[u|z]) ← sambung(x,y,z).
```

Takrifkan, dengan menggunakan "sambung", predikat-predikat di bawah:

- (i) `singkir(u,x,y)` : benar jika senarai y adalah sama dengan senarai yang didapati jika unsur u (yang pertama, jika lebih dari satu) disingkirkan daripada senarai x.

contoh:

```
singkir(2,[1,2,4,5,2], [1,4,5,2]) benar
```

- (ii) `palindrom(x)` : benar jika senarai x ialah suatu palindrom.

contoh:

```
palindrom ( 1,2,3,2,1 ) benar  
palindrom ( a,b,c,c,b,a ) benar
```

(40/100)

- (c) Suatu predikat "apakah_aku(x,y,z)", di mana x,y dan z adalah senarai, ditakrifkan seperti berikut:

```
apakah_aku([],y,[]).  
apakah_aku([u|x],y,z) ←  
    sambung(h,[u|r],y) apakah_aku(x,y,z).  
apakah_aku([u|x],y,[u|z]) ←  
    not sambung(h,[u|r],y), apakah_aku(x,y,z).
```

di mana "sambung" tertakrif seperti di atas. perhubungan apakah yang ditakrifkan oleh "apakah_aku"? Berikan contoh-contoh x,y, dan z yang menjadikan "apakah_aku" benar.

(35/100)

5. (a) Jelaskan bagaimana senarai sekutu diguna untuk mengimplementasikan kawasan ikatan bagi sesuatu fungsi. Tunjuk, terutama sekali, bagaimana kawasan ikatan untuk fungsi rekursif disusun supaya nilai ikatan pembolehubah tempatan dapat dicapai dengan betul.

(30/100)

- (b) Telitikan takrifan fungsi berikut:

```
(DEFINE (ENTAH_APA X E)
  (COND
    ( (NULL X) NIL)
    ( T (CONS
        (CONS (CAR X) (CONS E NIL))
        (ENTAH_APA (CDR X) E)
        ))))
```

Apakah yang dihitungkan oleh "ENTAH_APA". Berikan contoh-contoh input serta hasil ENTAH_APA.

(30/100)

- (c) (i) Jelaskan bagaimana suatu senarai boleh mewakili struktur pohon am.
- (ii) Menganggapkan sesuatu pohon itu diwakili seperti yang anda jelaskan, tuliskan suatu fungsi LISP yang akan menentukan bilangan nod daun (i.i. nod yang tidak ada anak) yang terdapat di dalamnya.

(40/100)

...ooOoo...