

UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Pertama  
Sidang Akademik 1997/98

September 1997

**CSI502 - Penyelesaian Masalah dan Pengaturcaraan**

Masa : [3 jam]

---

**ARAHAN KEPADA CALON:**

- Sila pastikan bahawa kertas peperiksaan ini mengandungi **EMPAT** soalan di dalam **ENAM** muka surat yang bercetak sebelum anda memulakan peperiksaan ini.
  - Jawab **SEMUA** soalan. Anda boleh memilih untuk menjawab semua soalan dalam Bahasa Malaysia atau Bahasa Inggeris.
  - Gunakan bahasa pengaturcaraan C di mana perlu.
-

1. (a) (i) Terangkan secara ringkas 4 peringkat utama dalam proses penyelesaian masalah menggunakan komputer .
- (ii) Berikan 2 prinsip-prinsip untuk mencapai reka bentuk atur cara yang berketumpatan (cohesion) tinggi.

[30/100]

- (b) Anda dikehendaki mereka bentuk satu atur cara yang menerima tiga nombor dan menyusun nombor-nombor tersebut supaya nombor yang nilainya paling kecil disimpan di NOM1, nombor yang nilainya kedua besar disimpan di NOM2, dan nombor yang nilainya paling besar disimpan di NOM3.

- (i) Bincangkan keperluan data dari segi input dan output.
- (ii) Nyatakan modul (-modul) yang terlibat. Modul (-modul) ini mestilah menerangkan submasalah bagi atur cara di atas.
- (iii) Tuliskan algoritma yang lengkap untuk modul (-modul) yang telah anda kenalpasti di bahagian (ii).

[40/100]

- (c) (i) Tuliskan prototaip fungsi dalam bahasa C bagi penerangan di bawah:

Fungsi **test** akan menerima tiga argumen, argumen pertama merupakan satu tatasusunan berjenis **float**, argumen kedua merupakan satu **integer** dan argumen ketiga adalah satu penuding kepada data berjenis **double**. Fungsi **test** akan memulangkan satu data berjenis penuding kepada **integer**.

- (ii) Jelaskan penggunaan utama prototaip fungsi dalam bahasa C.

[20/100]

- (d) Berdasarkan pengisytiharan pemboleh-pemboleh ubah di bawah, apakah output yang akan dicetak oleh setiap pernyataan C berikut (catatan: setiap soalan adalah bebas dari satu sama lain)

```
int x = 0, y = 10, z = 20;
```

- (i) `z++;`  
`x += y++ + z;`  
`printf (" x = %d, y = %d, z = %d\n", x, y, z);`
- (ii) `if (!x && y <= z)`  
`printf (" x = %d, y = %d, z = %d\n", ++x, y*10, z-10);`  
`else`  
`printf (" x = %d, y = %d, z = %d\n", x+10, y--, ++z);`
- (iii) `x = y * z / 5 * 4;`  
`printf (" x = %d\n", x);`

[10/100]

2. (a) Diberi pengisytiharan tatasusunan seperti di bawah, tuliskan satu fungsi (bagi setiap bahagian (i) dan (ii)) yang akan menokok nilai 2 kepada setiap elemen tatasusunan **x**.

```
int x[5] = { 10, 30, 50, 70 }
```

- (i) Panggilan menerusi nilai.
- (ii) Panggilan menerusi rujukan.
- (iii) Jelaskan mengapa kadang kala ada baiknya memanggil fungsi menerusi rujukan? Huraikan jawapan anda berdasarkan dengan merujuk kepada contoh pada soalan 2 (a) (i) dan 2 (a) (ii).

[25/100]

- (b) Anda ditugaskan untuk menguji fungsi **ComputeGrade** yang mengira gred peperiksaan (A-D) bersesuaian dengan markah yang diberikan (dalam julat 0 hingga 100). Markah lebih besar atau sama dengan 70 akan diberikan gred A, markah dari 50 hingga 69 akan diberikan gred B, markah dari 30 hingga 49 akan diberikan gred C dan markah lebih kecil atau sama dengan 30 akan diberikan gred D. Spesifikasi fungsi ini adalah seperti berikut:

Nama:	ComputeGrade		
Argumen-argumen:			
IN	Mark		markah mentah (satu integer)
OUT	Grade		Gred yang bersesuaian
OUT	OK		Benar jika markah adalah sah (dalam julat 0 - 100)

Penerangan: Jika **Mark** adalah sah, **OK** akan mempunyai nilai benar dan **Grade** akan mengandungi nilai gred yang bersesuaian dengan **Mark**. Jika markah tidak sah, **OK** akan mengandungi nilai palsu dan nilai yang terdapat pada **Grade** tidak diketahui.

Kesan Sampingan: tiada.

- (i) Tuliskan fungsi berdasarkan spesifikasi yang diberikan di atas.
- (ii) Kenal pasti kes-kes ujian untuk menguji fungsi tersebut. Sediakan pelan ujian dalam bentuk jadual yang mengandungi data-data seperti data ujian, tujuan (kes-kes ujian) dan keputusan yang dijangkakan

[30/100]

- (c) Berdasarkan pengisytiharan pemboleh-pemboleh ubah berikut, nyatakan output yang akan dicetak oleh setiap pernyataan C di bawah. Pernyataan-pernyataan C dalam setiap soalan tidak berkaitan antara satu sama lain.

```
char name[] = "CSI502 - Principles of Programming";
char *nPtr, ch;
char addr[80];
```

- (i) `nPtr = name;`  
`printf ("%s", nPtr);`
- (ii) `nPtr = name; ch = *nPtr;`  
`printf ("%c", ch);`

- (iii) `nPtr = name + 9; strcpy (addr, nPtr);  
printf ("%s", addr);`
- (iv) `nPtr = name + 6; *nPtr = '\0';  
printf ("%s",name);`
- (v) `nPtr = name;  
for (; (ch = *nPtr++)!= '\0';)  
printf ("%c",ch);`

[10/100]

(d) Tuliskan pernyataan gelung yang akan mengira jumlah setiap integer ketiga, bermula dari nilai  $i = 2$ , (iaitu kira jumlah  $2+5+8+11+...$ ) untuk semua nilai  $i$  yang kurang dari 100. Tuliskan gelung bagi tiga kaedah yang berbeza.

- (i) Menggunakan gelung kawalan **while**.
- (ii) Menggunakan gelung kawalan **do - while**.
- (iii) Menggunakan gelung kawalan **for**.

[20/100]

(e) (i) Apakah output yang akan dihasilkan oleh atur cara berikut?

```
#include <stdio.h>

void a (int);
void b (void);
void c (void);

int y = 5;

main() {
    int y = 10;

    a (y);
    b ();
    printf ("main --> %d\n",y);
}

void a (int y) {
    int x = 10;

    printf ("a --> %d\n",x+y);
    y += 20;
    c();
}

void b () {
    y *= 2;
    printf ("b --> %d\n",y);
    c();
}

void c () {
    static int y = 30;

    printf ("c --> %d\n",y);
    y += 20;
}
```

- (ii) Apakah yang dilakukan oleh prapemproses **#include** dan **#define**? Berikan contoh yang bersesuaian untuk menyokong jawapan anda. [15/100]
3. (a) Beri definisi struktur yang mempunyai ahli-ahli seperti berikut:
- satu rentetan bersaiz 80 aksara yang dipanggil **name**.
  - satu kuantiti **integer** yang dikenali **id**.
  - satu penuding rujuk sendiri yang diberi nama **link**.
- [15/100]
- (b) Dengan menggunakan **typedef**, definisikan satu jenis data baru bagi struktur dalam bahagian 3 (a) di atas dan namakannya sebagai **patient**. [5/100]
- (c) Isytiharkan pemboleh-pemboleh ubah berikut:
- Satu penuding kepada data berjenis **patient** dan namakannya sebagai **p\_start**.
  - Satu pemboleh ubah kepada data berjenis **patient** dan namakannya sebagai **patient1**.
- [10/100]
- (d) Dengan menggunakan pemboleh-pemboleh ubah yang telah anda isytiharkan dalam bahagian 3 (c), tuliskan pernyataan-pernyataan C yang bersesuaian untuk melakukan tugas berikut:
- (i) Nilai awalkan kandungan pemboleh ubah **patient1** dengan nilai-nilai berikut:
- ahli **name** dengan rentetan "Bailey".
  - ahli **id** dengan nilai 1007.
  - ahli **link** dengan nilai **NULL**.
- (ii) Nilai awalkan kandungan data yang dirujuk oleh pemboleh ubah **p\_start** dengan nilai-nilai yang sama seperti yang terdapat dalam bahagian 3 (d)(i). [10/100]
- (e) Tuliskan atur cara C yang akan melakukan tugas berikut:
- Atur cara ini akan menjana satu senarai berpaut yang mengandungi data-data pesakit. Atur cara ini akan memaparkan arahan-arahan yang bersesuaian kepada pengguna supaya pengguna dapat memasukkan data-data pesakit. Data-data tersebut akan disimpan dalam struktur data pesakit, dan kemudiannya akan diselitkan ke dalam senarai berpaut. Penyelitan mesti dilakukan mengikut turutan nama pesakit yang menaik. Atur cara ini juga mestilah memberi peluang kepada pengguna untuk berhenti atau meneruskan proses kemasukan data pesakit. Duplikasi rekod-rekod pesakit dibenarkan.
- Catatan: Gunakan struktur data **patient** yang telah anda definisikan dalam bahagian 3 (a) di atas. [40/100]

- (f) Jelaskan perbezaan konsep-konsep berikut:
- (i) Tindanan (Stack) and Baris Gilir (Queue).
  - (ii) Struktur data statik dan struktur data dinamik. [20/100]
4. (a) Tuliskan satu atur cara C yang membuka dan membaca fail capaian rawak **story.txt** secara berjjukan. Seterusnya atur cara ini akan memaparkan isi kandungan fail tersebut kepada output piawai. Fail **story.txt** mengandungi beberapa teks cerita pendek, dan setiap teks cerita pendek mempunyai saiz tetap sejumlah 80 aksara. [30/100]
- (b) Terangkan kelebihan-kelebihan dan kekurangan-kekurangan fail capaian rawak berbanding dengan fail berjjukan. [20/100]
- (c) Nyatakan beberapa sebab nilai NULL dipulangkan apabila anda membuka fail menggunakan fungsi **fopen**? [10/100]
- (d) (i) Terangkan apa yang dimaksudkan konsep-konsep orientasi objek berikut:
- Polimorfisme
  - Penyorokan data (Information Hiding)
  - Perwarisan (Inheritance)
- [20/100]
- (ii) Apa yang membezakan kelas dan objek? [10/100]
- (iii) Bagaimana penggunaan semula kod boleh dicapai dalam bahasa berorientasi objek? [10/100]