

UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Pertama
Sidang Akademik 2000/2001

September/Oktober 2000

CPT101/CAP101/CMP101 – Prinsip-Prinsip Pengaturcaraan

Masa : [3 jam]

ARAHAN KEPADA CALON:

- Sila pastikan bahawa kertas peperiksaan ini mengandungi **EMPAT** soalan di dalam **SEMBILAN** muka surat yang bercetak sebelum anda memulakan peperiksaan ini.
 - Jawab **SEMUA** soalan dalam Bahasa Malaysia.
 - Tentukan jawapan setiap soalan dimulakan pada muka surat baru.
 - Semua atur cara adalah dalam bahasa pengaturcaraan C++.
-

1. (a) Nyatakan sama ada kenyataan-kenyataan berikut BENAR atau PALSU.
- (i) Gelung ulangan bagi kenyataan **do-while** akan diuji pada bahagian akhir gelung tersebut.
 - (ii) Jenis pemboleh ubah dalam parameter sebenar tidak perlu sama dengan parameter formal dalam takrifan fungsi.
 - (iii) Struktur kawalan ulangan **for** lebih sesuai digunakan dalam keadaan di mana bilangan ulangan telah dikenal pasti terlebih dahulu.
 - (iv) Nilai awal data hanya boleh dilakukan semasa pengkompilan.
 - (v) Tidak semua struktur pilihan **if** sesuai ditulis semula dengan menggunakan struktur pilihan **switch**.

(10/100)

- (b) Jelaskan dengan ringkas ungkapan-ungkapan berikut:

- (i) pseudokod
- (ii) penterjemah bahasa
- (iii) mod komputer
- (iv) atur cara bermodul
- (v) kata kunci

(30/100)

- (c) Nyatakan output yang akan dihasilkan oleh cebisan-cebisan atur cara berikut:

```
(i) void main (){
    int kira (int, int);
    int i, j;
    for (i = 1; i < 4; i++) {
        j = kira(i, i + i);
        cout << "i = " << i << " j = " << j;
    }
}
int kira (int i, int j) {
    int jum = 0, k;
    for (k = i; k <= j; k++)
        jum += k;
    return jum;
}
```

```

(ii)   int x = 1;
       do {
           cout << x << endl;
           x = x + 3;
       } while (x != 10);

(iii)  void main () {
           cout << apo(5,6,'*');
           cout << apo(12,18,'^');
       }

int apo (int p, int q, char c) {
    int n = 0, i, j = 1;
    switch (c) {
        case '+' : for (i = p; i <= q; i++)
                    n += q;
                    return n;
                    break;
        case '*' : for (i = p; i <= q; i++)
                    j *= i;
                    return j;
                    break;
        case '^' : for (i = 0; i <= q; i++)
                    j = j * p;
                    return j;
                    break;
    }
}

```

(30/100)

(d) Kenal pasti dan betulkan ralat sintaks/ralat logik bagi setiap cebisan atur cara di bawah:

```

(i)   if (umur >= 40);
       cout << "Umur adalah lebih besar atau sama dengan 40";
       else if
       cout << "Umur kurang daripada 40;

(ii)  int x = 1;
       jum;
       while ( x <= 10)
           jum1 +=x;
           ++x;

(iii) while ( y > 0 ) {
           cout << y << endl;
           ++y;
       }

```

- (iv) **for** (x =100, x >=1, x/= 2)
 cout <<x <<endl;
- (v) **switch** (value % 2) {
 case '0':
 cout << "nombor genap \n";
 case '1':
 cout << "nombor ganjil \n";
 }

(30/100)

2. (a) Atur cara C++ berikut membuat analisis tentang satu keputusan ujian bagi 10 pelajar:

```

main ()
{
    int lulus = 0, gagal = 0, pelajar = 1, keputusan;

    while (pelajar <= 10) {
        cout << "Masukkan keputusan ujian ( 1=lulus, 2=gagal) : ";
        cin >> keputusan;

        if (keputusan == 1)
            lulus = lulus + 1;
        else
            gagal = gagal + 1;

        pelajar = pelajar + 1;
    }

    cout << "lulus = " << lulus << endl;
    cout << "gagal = " << gagal << endl;
}

```

Ubahsuai atur cara di atas dengan menambahkan ciri-ciri berikut :

- (i) Bagi sebarang data input yang dimasukkan, jika nilai yang dimasukkan selain daripada 1 atau 2, ulangi pertanyaan kemasukan data sehingga pengguna memasukkan nilai yang betul.
- (ii) Ubah kenyataan umpukan yang berkenaan kepada kenyataan umpukan menggunakan pengendali umpukan tokok ataupun susut.
- (iii) Baiki atur cara supaya boleh membuat analisis bagi **n** pelajar.

(25/100)

(b) Nombor perdana adalah merupakan satu nombor yang hanya boleh dibahagikan oleh dirinya sendiri sahaja. Contoh nombor perdana adalah 2, 3, 5, 11, 47 dan sebagainya.

(i) Tulis satu fungsi **perdana()** yang akan menerima satu nombor integer sebagai input, tentukan sama ada nombor tersebut adalah perdana ataupun tidak, dengan memulangkan nilai 1 atau 0 (benar atau palsu) kepada atur cara yang memanggilnya.

(ii) Tunjuk kenyataan yang digunakan untuk memanggil fungsi tersebut serta jenis semua pemboleh ubah yang digunakan.

(30/100)

(c) Sebuah syarikat letak kereta MSU mengenakan bayaran minimum RM2.00 bagi meletakkan kereta tiga jam pertama. Bayaran tambahan sebanyak RM0.50 dikenakan bagi setiap jam yang berikutnya (jika kurang daripada satu jam, bayaran bagi satu jam dikira, sebagai contoh 25 minit dikira 1 jam, begitu juga 45 minit dikira 1 jam). Bayaran maksimum bagi 24 jam ialah RM10.00. Anggapkan tiada siapa meletakkan kereta lebih dari 24 jam.

(i) Tulis pseudokod untuk mengira bayaran bagi tiga pelanggan yang meletakkan kereta mereka semalam. Input ialah bilangan jam dan minit meletak kereta untuk setiap pelanggan. Output ialah jumlah bayaran yang dikenakan.

(ii) Merujuk kepada 2(c)(i), tulis atur cara bemodul yang sekurang-kurangnya mempunyai dua fungsi iaitu fungsi **main** dan **kirabayaran**.

(45/100)

3. (a) Berikan jawapan ringkas untuk soalan-soalan berikut:

- (i) Pemulangan lebih daripada satu nilai daripada fungsi boleh dilakukan dengan menggunakan _____.
- (ii) Nama tatasusunan ialah _____.
- (iii) Struktur membolehkan penggabungan data-data _____.
- (iv) Alamat sesuatu pemboleh dan pemboleh aliasnya (pemboleh ubah rujukan) adalah _____.
- (v) Penyemakan sama ada sesuatu fail dapat dibuka dengan jayanya atau sebaliknya boleh dilakukan dengan menggunakan arahan _____.

(20/100)

(b) Perhatikan atur cara berikut dan jawab soalan-soalan selepasnya.

```
# include <iostream.h>
main ( )
{
    void kira (int, int *);
    int a = 5, b;

    kira (a, &b);
}

void kira (int x, int * y)
{
    * y = x + x;
}
```

- (i) Ubahsuai atur cara di atas supaya fungsi kira memulangkan nilai dengan menggunakan pemboleh ubah rujukan.
- (ii) Ubahsuai atur cara di atas supaya fungsi kira memulangkan nilai dengan menggunakan **return**.

(30/100)

- (c) Atur cara berikut mensimulasikan pembalangan dadu sebanyak seribu kali. Anda dikehendaki melengkapkan prototaip fungsi dan panggilan fungsi dan mentakrif ketiga-tiga fungsi tersebut.

```
# include <iostream.h>
# include <stdlib.h>
main ()
{
    void jana ( _____ ); // lengkapkan prototaip fungsi ini
    void simpan ( _____ ); // lengkapkan prototaip fungsi ini
    void cetak ( _____ ); // lengkapkan prototaip fungsi ini
    int i;
    for (i = 1, i <= 1000; i++) {
        jana ( _____ ); // lengkapkan panggilan fungsi ini
        simpan ( _____ ); // lengkapkan panggilan fungsi ini
    }
    cetak ( _____ ); // lengkapkan panggilan fungsi ini
}
```

/* Fungsi jana akan menjana satu nombor di antara 1 dengan 6 secara rawak (dengan menggunakan fungsi rand) setiap kali ia dipanggil dan memulangkan nilai tersebut kepada fungsi yang memanggilnya. */

/* Fungsi simpan menyimpan bilangan sesuatu nombor itu dihasilkan oleh pembalangan dadu. Bilangan ini perlu disimpan di dalam satu tatasusunan. */

/* Fungsi cetak akan mencetak bilangan terakhir untuk pembalangan yang sudah disimulasikan seperti berikut:

Nombor dadu	Bilangan
1	...
2	...
3	...
4	...
5	...
6	...

(50/100)

4. (a) (i) Takrifkan satu struktur untuk menyimpan maklumat-maklumat kursus seperti berikut:
- kod kursus
 - tajuk kursus
 - bilangan unit
- (ii) Isytiharkan satu pemboleh ubah untuk menyimpan maklumat lima kursus dengan menggunakan struktur yang ditakrifkan di dalam bahagian 4a(i).
- (iii) Tuliskan cebisan atur cara untuk mencetak kandungan pemboleh ubah yang diisytiharkan di dalam bahagian 4a(ii).

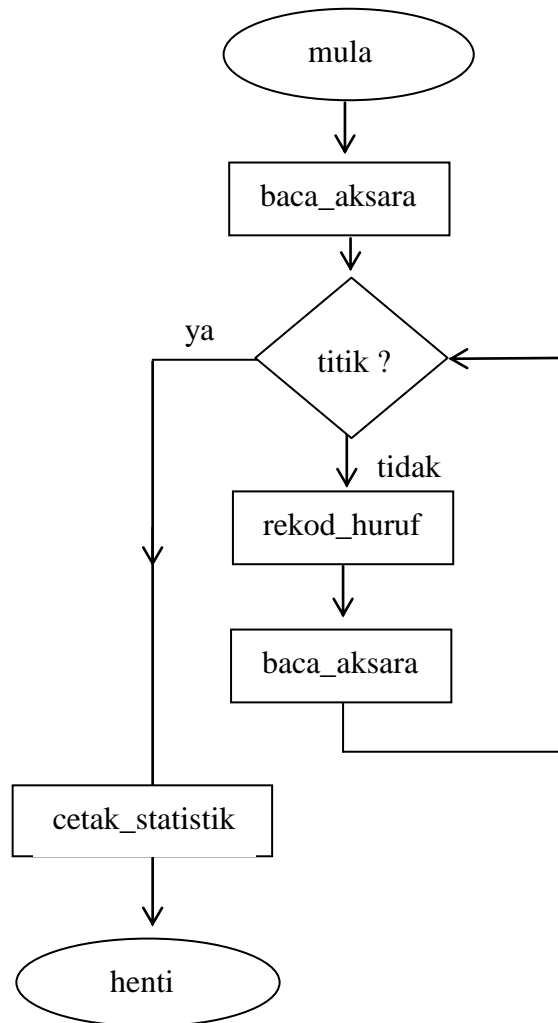
(20/100)

- (b) (i) Tulis semula struktur dari soalan bahagian 4(a) sebagai satu kelas.
- (ii) Terangkan perbezaan antara struktur dan kelas.
- (iii) Tuliskan satu fungsi ahli untuk mencetak peringkat kursus (iaitu kursus tahun 1, 2, 3 atau sarjana. Ini bergantung kepada nilai integer pertama dalam kod kursus).

Nilai integer pertama	Peringkat
1	Tahun pertama
2	Tahun kedua
3	Tahun ketiga
4 atau lebih	Sarjana

(20/100)

(c) Perhatikan carta aliran berikut:



Tuliskan satu atur cara berpandukan carta aliran di atas. Atur cara berkenaan akan membaca satu ayat. Sambil ia membaca, ia akan mengira bilangan setiap huruf yang ada di dalam ayat tersebut. Huruf besar dan kecil dianggap sama (contoh: a dan A). Proses membaca akan berhenti apabila titik dibaca. Selepas itu, bilangan setiap huruf akan dicetak.

Atur cara anda mesti mempunyai sekurang-kurangnya tiga fungsi:

- (i) **baca_aksara** – fungsi ini akan membaca aksara-aksara dari satu fail.
- (ii) **rekod_huruf** – fungsi ini akan mengemas kini bilangan huruf.
- (iii) **cetak_statistik** – fungsi ini akan mencetak bilangan huruf-huruf (a – z).

(60/100)