

UNIVERSITI SAINS MALAYSIA

Peperiksaan Kursus Semasa Cuti Panjang
Sidang Akademik 1998/99

April 1999

CAP102/CMP102 - Pengaturcaraan Lanjutan dan Struktur Data

CSC122 - 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 dalam Bahasa Malaysia.
-

1. Diberikan fungsi berikut:

```
void list_operation (head_node *list, int value)
{
    boolean    found;
    list_node  *curr_ptr, *prev_ptr;

    curr_ptr = list → head;
    prev_ptr = NULL;
    found = FALSE;

    while ((curr_ptr != NULL) && (!found)) {
        if (curr_ptr → id != value) {
            prev_ptr = curr_ptr;
            curr_ptr = curr_ptr → next;
        } else {
            found = TRUE
            if (prev_ptr == NULL) {
                list → head = curr_ptr → next;
            } else {
                prev_ptr → next = curr_ptr → next
            }
            free (curr_ptr);
        }
    }
    if (!found)
        printf ("\n*****");
}
```

(a) Apakah yang dilakukan oleh fungsi di atas? Jelaskan langkah demi langkah operasi yang akan dilaksanakan berdasarkan kod di atas (beri lakaran jika perlu). (5/100)

(b) Apakah yang berlaku sekiranya penuding `prev_ptr` dan `curr_ptr` dalam pernyataan `if` yang pertama ditukar tertib seperti berikut:

```
curr_ptr = curr_ptr → next;
prev_ptr = curr_ptr;
```

(5/100)

(c) Terangkan apa yang berlaku apabila fungsi di atas dilaksanakan ke atas senarai `list` yang kosong. (5/100)

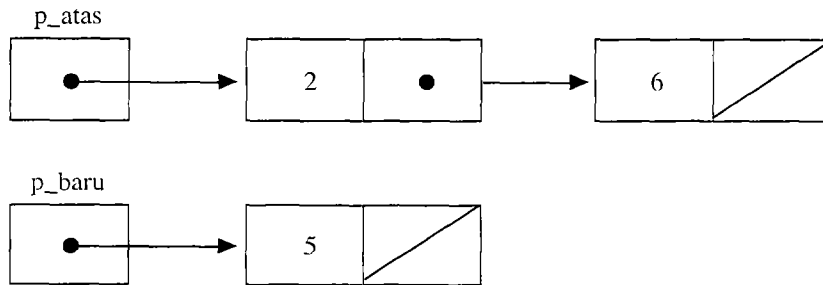
(d) Terangkan apa yang berlaku apabila nilai `value` yang dihantar kepada fungsi di atas tidak wujud di dalam senarai `list`. (5/100)

(e) Nyatakan kesan perlaksanaan fungsi di atas jika syarat `curr_ptr != NULL` di dalam arahan `while` digantikan dengan `curr_ptr->next != NULL`? (5/100)

2. (a) Beri perbezaan antara senarai, tindanan dan giliran.

(3/100)

(b) Andaikan wujud satu tindanan (stack) bersama satu nod baru yang hendak disisip seperti berikut: (Penuding p_atas menuding kepada nod pertama di dalam tindanan, manakala penuding p_baru menuding kepada nod yang hendak diselit.)



- (i) Berikan pengisytiharaan yang sesuai untuk tindanan di atas.
- (ii) Tunjukkan melalui lakaran, apakah perubahan yang berlaku langkah demi langkah apabila nod yang dituding oleh p_baru disisip ke dalam tindanan tersebut.
- (iii) Dua arahan diperlukan untuk proses penyisipan di atas. Tulis arahan-arahan tersebut.
- (iv) Bincangkan sama ada tertib kedua-dua arahan yang anda berikan dalam (ii) di atas penting di dalam proses penyelitan ini.

(12/100)

(c) Andaikan anda diminta untuk menggunakan suatu baris gilir keutamaan `PriorityQueue` yang cirinya berbeza daripada baris gilir biasa - iaitu `PriorityQueue` ini terisih mengikut keutamaan menurun berjenis integer. Anggapkan baris gilir ini dibina secara implementasi berpaut.

- (i) Nyatakan pengisytiharaan struktur data bagi `PriorityQueue` tersebut.
- (ii) Tulis suatu fungsi `CountGreater ()` yang akan menjejak `PriorityQueue` dan mengembalikan jumlah kunci yang lebih besar daripada nilai kunci *key*.
- (iii) Tulis suatu fungsi `PruneQueue ()` yang menghapuskan kesemua nilai yang lebih kecil atau sama dengan nilai kunci, *key*, yang diberikan. Anggapkan nilai lebih kecil (atau sama dengan kunci) sentiasa wujud.

(10/100)

3. (a) (i) Lukis pepohon gelintaran perduaan untuk jujukan nombor-nombor berikut:
 15 14 9 7 3 16 4 30 4 5
- (ii) Berikan hasil penyusunan pepohon perduaan yang dibina dalam soalan 3(a)(i) bagi kaedah berikut:
- Tertib Awalan
 - Tertib Akhiran
 - Tertib Sisipan

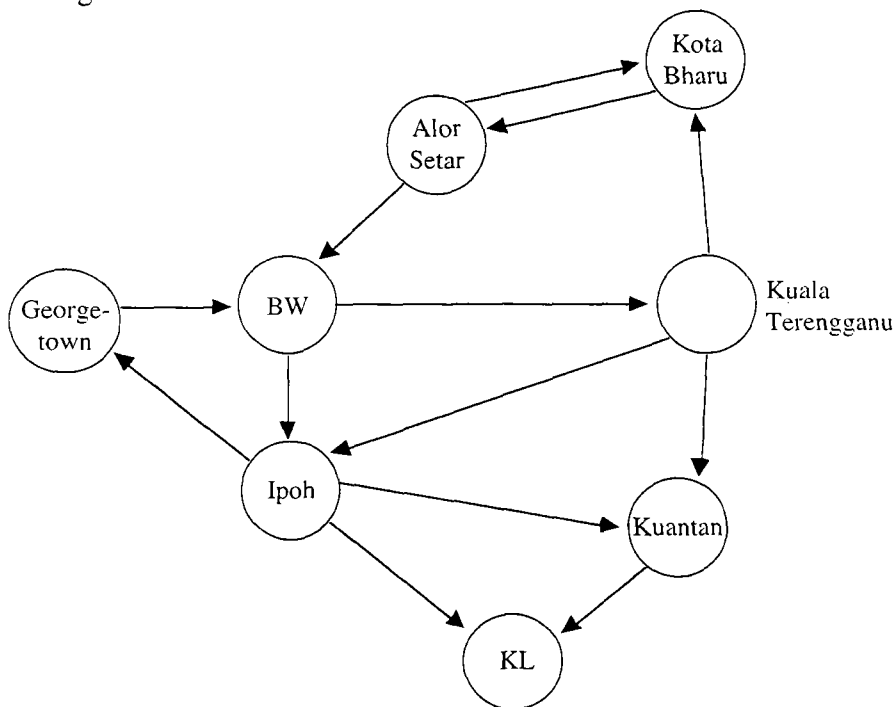
(11/100)

- (b) Pepohon gelintaran perduaan diwakilkan oleh satu set nod-nod. Setiap nod mempunyai sekurang-kurangnya 3 medan iaitu medan data, medan yang menuding kepada anak kiri dan medan yang menuding kepada anak kanan.

- (i) Berikan pengisytiharaan dalam bahasa C untuk implementasi di atas.
- (ii) Tulis satu fungsi yang akan menyenaraikan nod-nod secara terisih.

(8/100)

- (c) Diberi graf di bawah:



- (i) Beri implementasi graf sebagai:
- Jadual kesebelahan
 - Implementasi berpaut
- (ii) Tunjukkan langkah demi langkah untuk menjejak graf di atas menggunakan kaedah Gelintaran Kelebaran Dahulu (BFS) bermula dari Alor Setar. Tunjukkan kandungan baris gilir pada setiap langkah.

(6/100)

4. (a) Diberi jujukan nombor-nombor berikut:

30 62 53 42 17 97 91 38 51 18

Isih nombor-nombor di atas menggunakan mana-mana **tiga** dari empat algoritma di bawah:

- Isihan Shell
- Isihan Cepat
- Isihan Pilih
- Isihan Sisip

(12/100)

(b) Jika Isihan Cepat ingin diubahsuaikan untuk mengisih data secara menurun (bukan secara menaik), ubahsuaikan algoritma piawai berikut:

```
void QuickSort (List *list)
{
    RecQuickSort(list, 0, list → count-1);
}

void RecQuickSort (List *list, Position Low, Position High)
{
    Position pivotpos;
    if (low < high) {
        pivotpos = Partition (list, low, high);
        RecQuickSort (list, low, pivotpos -1);
        RecQuickSort (list, pivotpos +1, high);
    }
}

Position Partition (List *list, Position Low, Position High)
{
    ListEntry pivot;
    Position i, lastsmall, pivotpos;
    Swap(low, (low+high)/2, list);
    pivot = list->entry [low];
    pivotpost = low;
    for (i = low + 1; i <= high; i++)
        if (LT(list -> entry [i].key, pivot.key))
            Swap(++ pivotpos, i, list);
    Swap(low, pivotpos, list);
    return pivotpos;
}
```

Contoh output bagi algoritma yang diubahsuaikan:

Senarai asal → 5 7 4 9 11 8
 Senarai akhir → 11 9 8 7 5 4

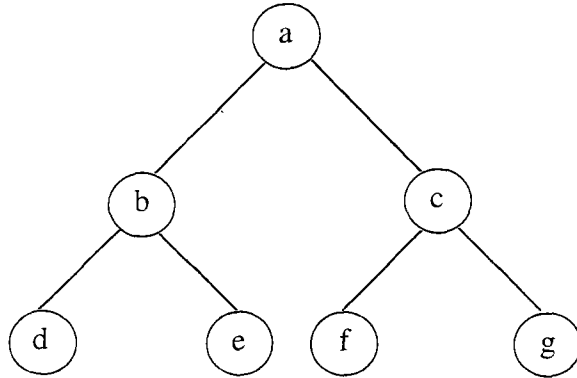
(5/100)

- (c) Tulis satu fungsi yang membuang semua nod-nod yang nilainya lebih besar daripada nilai X dalam pepohon gelintaran perduaan.

(4/100)

- (d) Tulis satu fungsi dalam Bahasa C yang akan menjejak pepohon paras demi paras.

Contoh:



Output: a b c d e f g

Anda dibenar menggunakan mana-mana struktur data abstrak yang anda telah pelajari.

(4/100)

- oooOooo -