UNIVERSITI SAINS MALAYSIA

Second Semester Examination
Academic Session 2007/2008

April 2008

MSG 383 – Data Structures for Computer Graphics
[Struktur Data untuk Grafik Komputer]

Duration : 3 hours
[Masa : 3 jam]

Please check that this examination paper consists of NINE pages of printed material before you begin the examination.

[Sila pastikan bahawa kertas peperiksaan ini mengandungi SEMBILAN muka surat yang bercetak sebelum anda memulakan peperiksaan ini.]

Instructions: Answer **all four** [4] questions.

[Arahan: *Jawab* ***semua empat*** *[4] soalan.]*

1. (a) Find the smallest and the largest unsigned integers that can be stored in 2 bytes.

(b) Suppose 16 bits are used to represent real numbers, where 8 bits store the mantissa and the other 8 bits store the exponent. Use twos-complement notation to find

(i) the smallest positive real number that can be represented by these 16 bits

(ii) the largest negative real number that can be represented by these 16 bits.

(c) Suppose 16 bits are used to represent real numbers, where the leftmost 8 bits store the mantissa and the next 8 bits store the exponent. Change the following numbers to twos-complement representations.

(i) 3600.0

(ii) -100.0×10^{-8}

(d) Use 8-bit twos-complement notation to carry out the following arithmetic calculations.

(i) $12+(-5)$

(ii) -3×4

[100 marks]

1. (a) *Cari nombor-nombor terkecil dan terbesar bagi integer tak bertanda yang dapat disimpan dalam 2 bait.*

(b) *Anggapkan 16 bit digunakan untuk mewakili nombor nyata, di mana 8 bit menyimpan mantisa dan 8 bit yang lain menyimpan eksponen. Gunakan tatatanda penggenap dua untuk mencari*

(i) nombor nyata positif terkecil yang boleh diwakili oleh 16 bit ini

(ii) nombor nyata negatif terbesar yang boleh diwakili oleh 16 bit ini.

(c) *Anggapkan 16 bit digunakan untuk mewakili nombor nyata, di mana 8 bit yang paling kiri menyimpan mantisa dan 8 bit yang seterusnya menyimpan eksponen. Tukarkan nombor-nombor berikut kepada perwakilan-perwakilan penggenap dua.*

(i) 3600.0

(ii) -100.0×10^{-8}

(d) *Gunakan 8-bit tatatanda penggenap dua untuk mengendali pengiraan aritmetik berikut.*

(i) $12+(-5)$

(ii) -3×4

[100 markah]

2. (a) Find the output of the following program :

```
#include <iostream.h>
void main()
{
int *p, *q, x = 2;
p = new int;
*p = 1;
q = p;
*q = x;
x = *p;
cout << "*p = " << *p << endl;
cout << "*q = " << *q << endl;
cout << "x = " << x << endl;
}
```

(b) What is wrong with the following declaration?

```
int *p = 2;
```

(c) State 3 differences between a static memory allocation and a dynamic memory allocation.

(d) The shuffle-merge of two lists $a_1, a_2, \ldots, a_m$, and $b_1, b_2, \ldots, b_n$, is the list

$a_1, b_1, a_2, b_2, \ldots, a_m, b_m, b_{m+1}, \ldots, b_n$ if $m < n$,

$a_1, b_1, a_2, b_2, \ldots, a_n, b_n, a_{n+1}, \ldots, a_m$ if $m > n$,

or

$a_1, b_1, a_2, b_2, \ldots, a_m, b_m$ if $m = n$.

Write an algorithm to shuffle-merge two linked lists with first nodes pointed to by `head1` and `head2`, respectively. The list elements in these two lists should be copied to produce the new list with first node pointed to by `head`. The original two lists should be destroyed.

[100 marks]

2. (a) *Cari output bagi program berikut :*

```
#include <iostream.h>
void main()
{
int *p, *q, x = 2;
p = new int;
*p = 1;
q = p;
*q = x;
x = *p;
cout << "*p = " << *p << endl;
cout << "*q = " << *q << endl;
cout << "x = " << x << endl;
}
```

(b) *Apakah kesalahan bagi pengisytiharan berikut?*

```
int *p = 2;
```

(c) *Nyatakan 3 perbezaan antara peruntukan ingatan statik dan peruntukan ingatan dinamik.*

(d) *Hasil pengocokan-cantum (shuffle-merge) bagi dua senarai* $a_1, a_2, \ldots, a_m$, *dan* $b_1, b_2, \ldots, b_n$, *ialah satu senarai*

$$a_1, b_1, a_2, b_2, \ldots, a_m, b_m, b_{m+1}, \ldots, b_n \qquad \text{jika } m < n,$$

$$a_1, b_1, a_2, b_2, \ldots, a_n, b_n, a_{n+1}, \ldots, a_m \qquad \text{jika } m > n,$$

atau

$$a_1, b_1, a_2, b_2, \ldots, a_m, b_m \qquad \text{jika } m = n.$$

Tuliskan satu algoritma untuk mengocok-cantum dua senarai berpaut yang ditunjuk oleh `head1` *dan* `head2` *masing-masing. Unsur-unsur dalam dua senarai ini mesti disalin kepada satu senarai baru yang ditunjuk oleh* `head`. *Dua senarai yang asal itu perlu dihapuskan.*

[100 markah]

3. (a) Refer to Figure 3.1 given below. Obtain the binary tree traversals in

(i) Preorder

(ii) Inorder

(iii) Postorder.

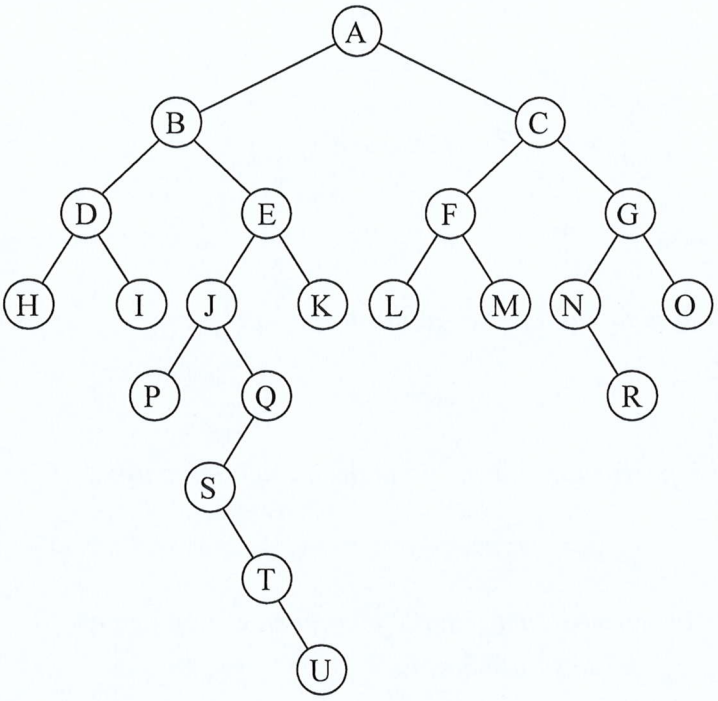


Figure 3.1

(b) Given a data file which contains the letters as below:

```
AAABBBAACCCCAADDD
EEEEDDDDDDDDBBBBB
BBBBBBDDDDDEEEEEE
EDDDEEEEEDDDDDDDD
DDDD
```

(i) Use the Huffman coding to build a Huffman's tree.

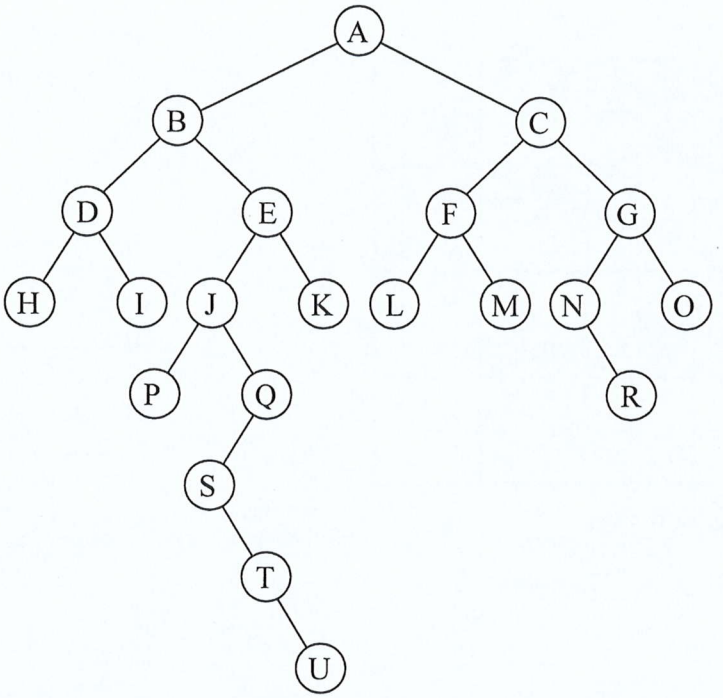
(ii) Construct the Huffman codes for each letter that existed in the file.

3. (a) *Rujuk kepada Gambarajah 3.1 di bawah. Dapatkan penyusuran pepohon perduaan dalam*

(i) Tertib awalan

(ii) Tertib sisipan

(iii) Tertib akhiran.



Gambarajah 3.1

(b) Diberi suatu fail data yang mengandungi huruf-huruf seperti berikut:

```
AAABBBAACCCCAADDD
EEEEDDDDDDDDBBBBB
BBBBBBDDDDDEEEEEE
EDDDEEEEEDDDDDDDD
DDDD
```

(i) Gunakan pengkodan Huffman untuk menbina satu pepohon Huffman.

(ii) Dapatkan kod-kod Huffman untuk setiap huruf yang wujud di dalam fail data.

3. (c) Figure 3.2 shows a picture that is divided into four quadrants in the order shown in Figure 3.3. Use the following notations

F : Full quadrant

P : Partially full quadrant

E : Empty quadrant

to picture a Quadtree of depth 3 for Figure 3.2. Label the branches and the nodes clearly.

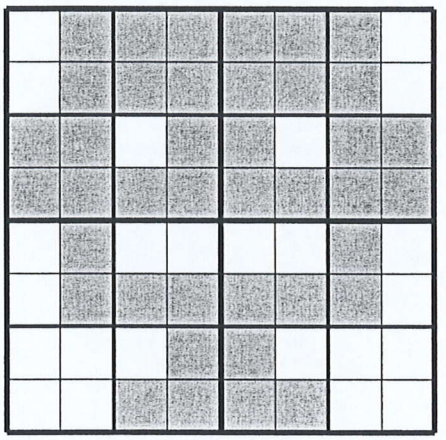


Figure 3.2

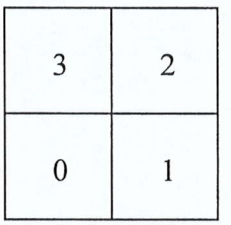


Figure 3.3

[100 marks]

4. (a) Write a non-recursive C++ function for the Bernstein polynomial

$$B_i^n(t) = \frac{n!}{i!\,(n-i)!}\, t^i\,(1-t)^{n-i}$$

where $t \in [0, 1]$, $n \geq 0$ and $i = 0, 1, \ldots, n$.

(b) Write a recursive C++ function for the Bernstein polynomial

$$B_i^n(t) = (1-t)\,B_i^{n-1}(t) + t\,B_{i-1}^{n-1}(t), \qquad \text{if } n > 0,$$

and $$B_i^0(t) = \begin{cases} 1 & \text{if } i = 0 \\ 0 & \text{if } i \neq 0. \end{cases}$$

[100 marks]

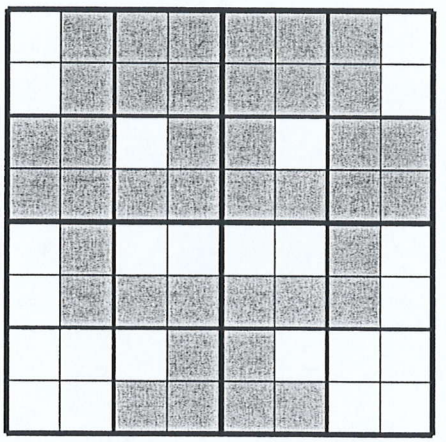
3. (c) *Gambarajah 3.2 memaparkan satu gambar yang dibahagi kepada empat sukuan dalam tertib yang ditunjukkan dalam Gambarajah 3.3. Gunakan tatatanda berikut*

F : Sukuan penuh

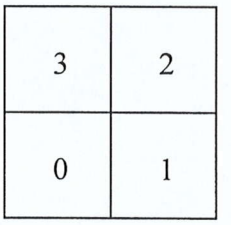
P : Sukuan penuh separa

E : Sukuan kosong

untuk memapar satu pepohon Kuad yang mempunyai kedalaman 3 bagi Gambarajah 3.2. Tandakan setiap laluan dan nodnya.



Gambarajah 3.2



Gambarajah 3.3

[100 markah]

4. (a) *Tuliskan satu fungsi C++ bercorak tak rekursi untuk polinomial Bernstein*

$$B_i^n(t) = \frac{n!}{i!\,(n-i)!}\, t^i\,(1-t)^{n-i}$$

dimana $t \in [0, 1]$, $n \geq 0$ *dan* $i = 0, 1, \ldots, n$.

(b) *Tuliskan satu fungsi C++ bercorak rekursi untuk polinomial Bernstein*

$$B_i^n(t) = (1-t)\,B_i^{n-1}(t) + t\,B_{i-1}^{n-1}(t), \qquad \textit{jika } n > 0,$$

dan
$$B_i^0(t) = \begin{cases} 1 & \textit{jika } i = 0 \\ 0 & \textit{jika } i \neq 0. \end{cases}$$

[100 markah]

- ooo O ooo -