

---

UNIVERSITI SAINS MALAYSIA

Second Semester Examination  
2011/2012 Academic Session

June 2012

**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 EIGHTEEN pages of printed material before you begin the examination.

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

**Instructions** : Answer **all six** [6] questions.

**Arahan** : Jawab **semua enam** [6] soalan.

The question papers **shall not be taken out** from the examination hall and will be collected by invigilators.

***Kertas soalan ini tidak boleh dibawa keluar daripada dewan peperiksaan dan akan dikutip oleh pengawas peperiksaan.***

In the event of any discrepancies, the English version shall be used.

*[Sekiranya terdapat sebarang percanggahan pada soalan peperiksaan, versi Bahasa Inggeris hendaklah diguna pakai].*

...2/-

1. Answer the following questions,

(a) Prove that if given  $n$  bit, the range of the integers that can be represented by

(i) one's complement notation is - to , and

(ii) two's complement notation is to .

*Hint:* find the upper and lower limits in the range for each notation by assuming  $n = 8$ .

(b) If  $m+n$  bit is used to represent a real number, where the first  $m$  bits represent mantissa and the last  $n$  bits represents exponents. Prove that,

(i) The range of the **positive** real numbers which can be represented by two's complement notation is to , and

(ii) The range of the **negative** real numbers which can be represented by two's complement notation is to .

[100 marks]

1. Jawab soalan-soalan berikut,

(a) *Buktikan bahawa jika diberi  $n$  bit, julat integer yang boleh diwakili oleh*

(i) *tatatanda penggenap satu ialah dari - ke , dan*

(ii) *tatatanda penggenap dua ialah dari ke .*

*Petua: cari had-had atas dan bawah untuk julat tatatanda masing-masing dengan menganggap bahawa  $n = 8$ .*

(b) *Jika  $m+n$  bit digunakan untuk mewakili satu nombor nyata, di mana  $m$  bit yang pertama mewakili mantisa dan  $n$  bit yang terakhir mewakili eksponen, buktikan bahawa,*

(i) *julat nombor nyata **positif** yang boleh diwakili oleh tatatanda penggenap dua ialah dari ke , dan*

(ii) *julat nombor nyata **negatif** yang boleh diwakili oleh tatatanda penggenap dua ialah dari ke .*

[100 markah]

2. Mark the following statements as **True** or **False**.

- (a) A variable created during program execution is called a dynamic variable.
- (b) If  $p$  is a pointer of the type `int`, then the statement `delete p`; de-allocates the memory pointed to by  $p$ .
- (c) A recursive algorithm solves a problem by reducing it to smaller versions of itself.
- (d) In recursion, a base case is a statement which calls itself recursively.
- (e) Recursion is the most efficient way of processing Fibonacci problem.
- (f) A linear search is not efficient for large lists compare to binary search. However, if a large list is sorted, the linear search is as efficient as the binary search.
- (g) In a quick sort, the sorting work is done by using “divide and conquer” technique to partition the list.
- (h) A binary tree must be nonempty.
- (i) The Inorder Traversal of a binary tree always outputs the data in ascending order.
- (j) The features of a fractal include self-similarity, irregular dimension and it has a recursive definition.

[100 marks]

2. Tandakan pernyataan-pernyataan berikut sebagai **True** atau **False**.

- (a) *Pembolehkan yang dicipta semasa pelaksanaan program dipanggil pembolehkan dinamik.*
- (b) *Jika  $p$  adalah suatu penuding dengan jenis `int`, pernyataan `delete p`; membebaskan peruntukan ingatan yang ditunjukkan oleh  $p$ .*
- (c) *Algoritma rekursi menyelesaikan suatu masalah dengan mengurangkan masalah tersebut kepada versi sendiri yang lebih kecil.*
- (d) *Dalam `recursi`, kes asas adalah pernyataan yang memanggil fungsi sendiri secara rekursi.*
- (e) *Rekursi adalah cara yang paling cekap untuk memproses masalah Fibonacci.*
- (f) *Gelintaran linear tidak cekap untuk memproses senarai yang besar jika dibandingkan dengan gelintaran perduaan. Tetapi, gelintaran linear adalah secepat gelintaran perduaan apabila memproses senarai besar yang terisih.*
- (g) *Isihan cepat menggunakan teknik “pecah dan perintah” untuk membahagikan senarai.*
- (h) *Suatu pepohon perduaan mestilah bukan kosong.*
- (i) *Penyusuri Tertib Sisipan untuk suatu pepohon perduaan selalu memberi output data dalam tertib menokok.*
- (j) *Ciri-ciri untuk suatu fraktal termasuk swakeserupaan, dimensi tidak tetap dan mempunyai definisi rekursi.*

[100 markah]

...4/-

3. Answer the following questions,

- (a) Describe Bubble Sort and Quick Sort algorithms.
- (b) Sort the following list with Bubble Sort and Quick Sort,

43, 56, 1, 7, 99, 43

- (i) For Bubble Sort, show the result of the sorting for each pass.
- (ii) For Quick Sort, use the first element as the pivot, show the new list whenever there is a swap of values in the list, and state the new pivot for the sub-lists.
- (c) On the sorted list produced in (b), describe how Binary Search can be used to search for a number. Describe both scenarios when the key number is found (use key = 7) and not found (use key = 20)

[100 marks]

3. *Jawab soalan-soalan berikut,*

- (a) *Terangkan algoritma-algoritma Isihan Buih dan Isihan Cepat.*
- (b) *Isihkan senari di bawah dengan Isihan Buih dan Isihan Cepat.*

43, 56, 1, 7, 99, 43

- (i) *Untuk Isihan Buih, tunjukkan hasil isihan untuk setiap laluan.*
- (ii) *Untuk Isihan Cepat, gunakan unsur pertama sebagai pangsi, tunjukkan senarai baru apabila terdapat pertukaran nilai-nilai dalam senarai, dan nyatakan pangsi baru untuk sub-senarai.*
- (c) *Pada senarai terisih yang dihasilkan di (b), terangkan bagaimana Gelintaran Perduaan boleh digunakan untuk mencari satu nombor. Terangkan kedua-dua scenario apabila kekunci dijumpa (guna kekunci = 7) dan tidak dijumpa (guna kekunci = 20).*

[100 markah]

...5/-

4. Answer the following questions,

- (a) Define Binary Tree.
- (b) Describe Preorder, Inorder and Postorder Traversals of a binary tree.
- (c) Given the data structure of a binary tree as in Figure 1, write recursive algorithms for the 3 traversals you described in (b).

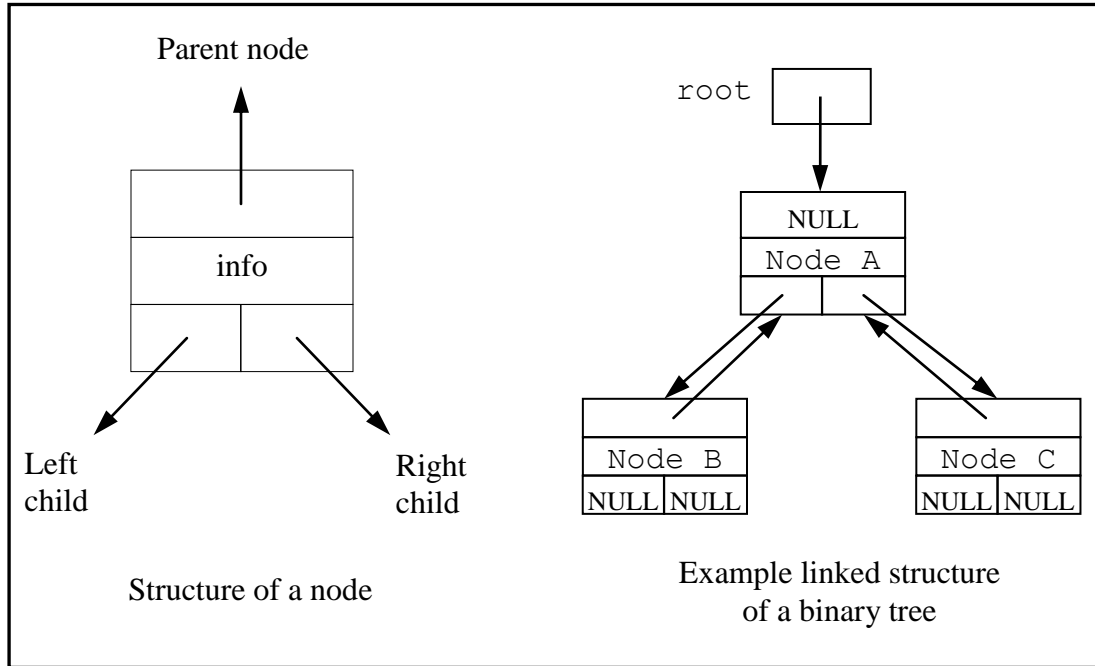


Figure 1

(d) The following code lists the nodes in a binary tree in two different orders:

Preorder: ABDEGHJKLMCFI

Inorder: DBGEHKJMLACIF

- (i) Construct (draw manually) the binary tree.
- (ii) Give the Postorder Traversal for the binary tree constructed.

[100 marks]

4. Jawab soalan-soalan berikut,

- (a) Takrifkan Pepohon Perduaan.
- (b) Huraikan Penyusuri Tertib Awalan, Sisipan dan Akhiran untuk suatu pepohon perduaan.
- (c) Diberi struktur data untuk suatu pepohon perduaan seperti di Gambarajah 1, tuliskan algoritma-algoritma rekursi untuk ketiga-tiga penyusuri yang kamu huraikan di (b).

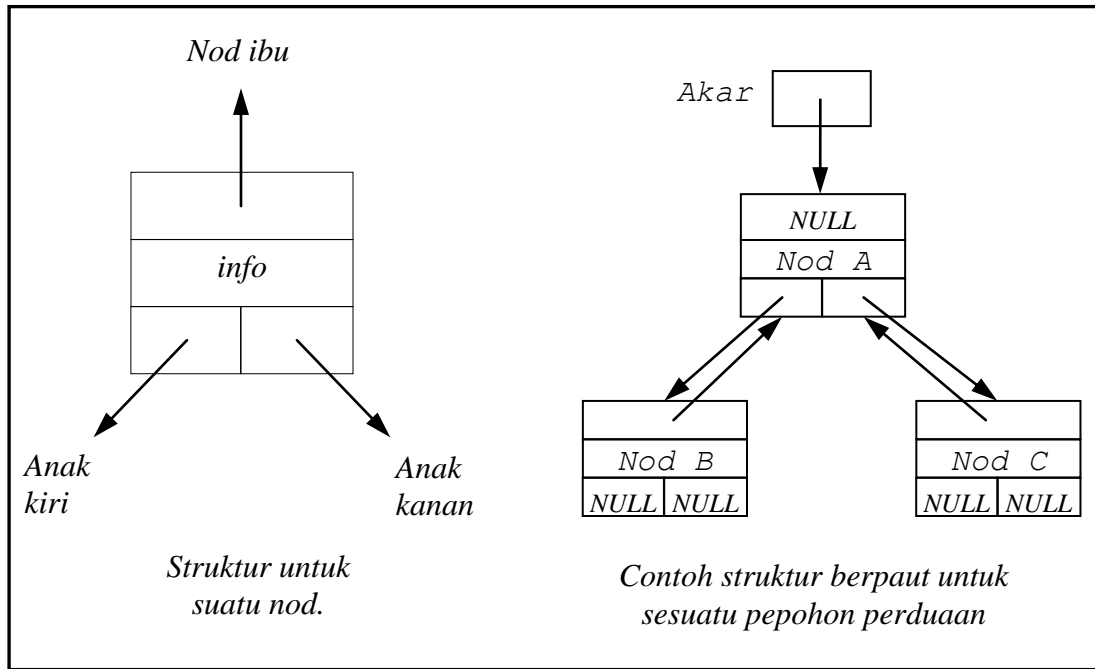


Figure 1

(d) Kod-kod di bawah senaraikan nod-nod untuk suatu pepohon perduaan dalam dua tertib yang berbeza:

Tertib awalan: ABDEGHJKLMCFI

Tertib sispan: DBGEHKJMLACIF

- (i) Binakan (lukis secara manual) pepohon perduaan tersebut.
- (ii) Berikan Tertib Akhiran pepohon perduaan tersebut.

[100 markah]

...7/-

5. By using Box-Counting Dimension method, let  $r$  be the side length of a fractal and  $N(r)$  be the number of boxes with side length  $r$ , which are needed to cover the shape. For each of the fractals A and B given in Figure 2, answer questions (a) to (c) as follows,

(a) find appropriate values of  $r$  and fill in the table below for  $n = \{1, 2, 3, 4, 5\}$ ,

$n$	$r_n$	$N(r_n)$	—
:	:	:	:

(b) find a pattern in the values of  $N(r_n)$ , and

(c) use this pattern to compute the box-counting dimension,  $d_b$ , which is given as

$$d_b = \frac{\quad}{\quad}$$

$$= \frac{\quad}{\quad}$$

If the calculated dimension  $d_b$  is not an integer number, leave your answer in the simplest Log form, i.e.  $d_b = \frac{\log(\quad)}{\log(\quad)}$ .

[100 marks]

5. Dengan menggunakan kaedah Dimensi Pengiraan-Kotak, biar  $r$  sebagai panjang tepi untuk suatu fraktal dan  $N(r)$  sebagai bilangan kotak, dengan panjang tepi,  $r$ , yang dikehendaki untuk menutup bentuk fraktal tersebut. Untuk setiap fraktal A dan B yang ditunjuk di Gambarajah 2, jawab soalan-soalan dari (a) ke (c) seperti berikut,

(a) cari nilai-nilai yang sesuai untuk  $r$  dan isikan jadual di bawah untuk  $n = \{1, 2, 3, 4, 5\}$ ,

$n$	$r_n$	$N(r_n)$	—
:	:	:	:

(b) cari satu corak dalam nilai-nilai  $N(r_n)$ , dan

(c) gunakan corak tersebut untuk mengira dimensi pengiraan-kotak,  $d_b$ , yang diberi sebagai

$$d_b = \frac{\quad}{\quad}$$

$$= \frac{\quad}{\quad}$$

Jika dimensi  $d_b$  yang dikira bukan suatu nombor integer, biarkan jawapan anda dalam bentuk Log, i.e.  $d_b = \frac{\log(\quad)}{\log(\quad)}$ , yang paling mudah.

[100 markah]

...8/-

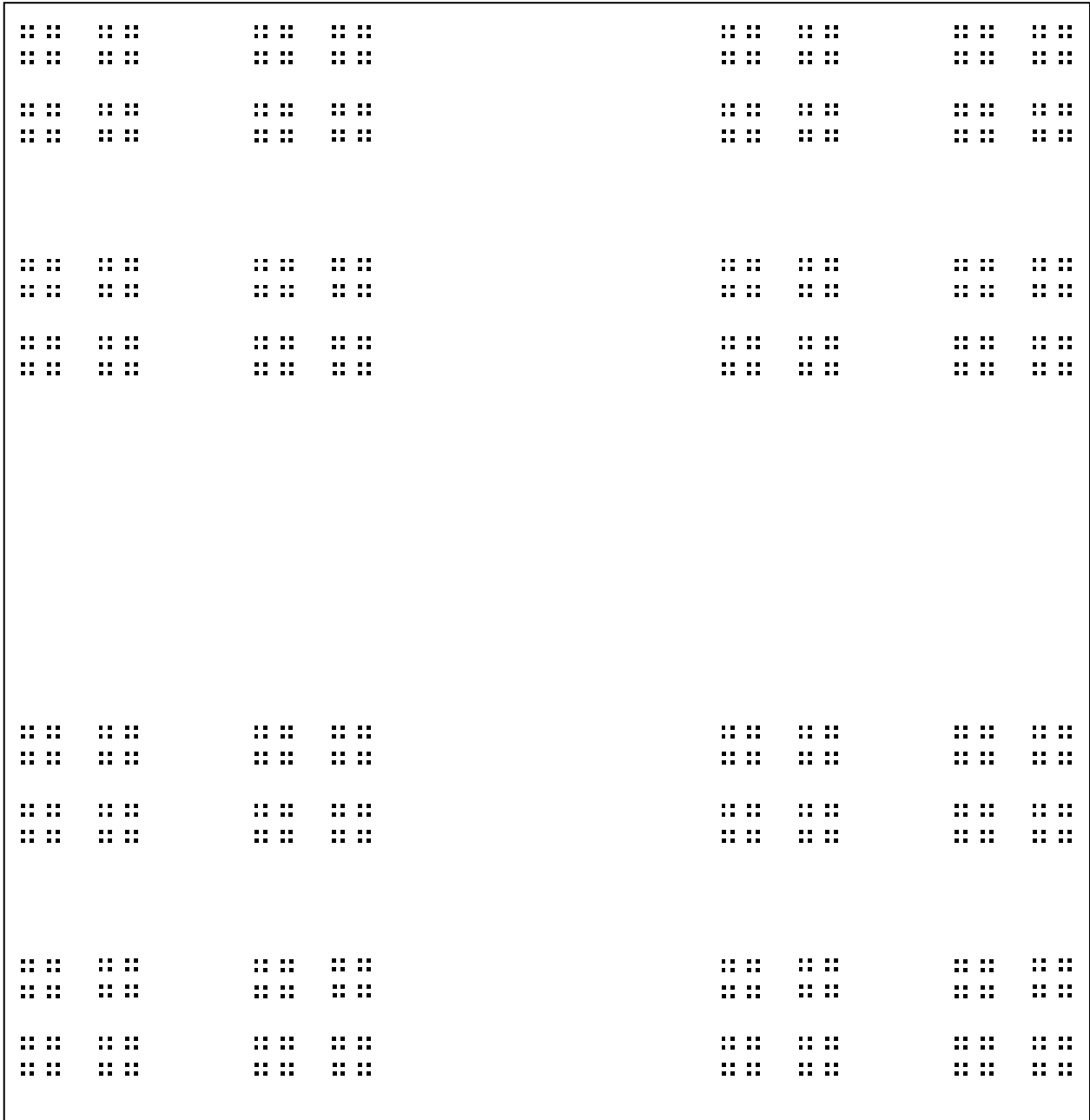


Figure 2.

*Gambarajah 2.*

...9/-

6. The following program segment show C++ array representations which store two Quadtrees for two images, A and B.

```
unsigned int A[] = {
    2,
    2, 2, 1, 0, 1, 0, 0, 2, 1, 0, 1, 0, 0,
    2, 0, 2, 0, 1, 0, 1, 0, 2, 0, 1, 0, 1,
    2, 2, 1, 0, 1, 0, 0, 2, 1, 0, 1, 0, 0,
    2, 0, 2, 0, 1, 0, 1, 0, 2, 0, 1, 0, 1,
};

unsigned int B[] = {
    2,
    2, 2, 0, 0, 1, 0, 1, 2, 1, 0, 0, 0, 1,
    2, 1, 2, 0, 0, 0, 1, 1, 2, 0, 1, 0, 0,
    2, 0, 1, 0, 1,
    2, 1, 0, 1, 0,
};
```

By referring to the above program segment, answer the following questions. For questions (b) – (e), use the answer sheets prepared in “Appendix: Answer Sheets for Question 6”. Detach your answer from this examination paper and attach to your answer booklet.

- (a) Convert the values of arrays A and B to binary codes with 2 bits.
- (b) From the binary codes in (a), draw the respective quadtrees for A and B. Label the nodes of the tree with the binary codes.
- (c) From the quadtrees in (b), shade the respective pixel representations for A and B.
- (d) From (c), find the pixel representations of A B and A B.
- (e) From (d), find the quadtrees of A B and A B. Label the nodes of the quadtrees with binary codes.
- (f) From (d) or (e), find the C++ array representations for A B and A B, up to level 3, i.e. as in program segment above for images A and B.

[100 marks]

...10/-

6. Segmen program di bawah menunjukkan perwakilan-perwakilan tatasusunan C++ yang menyimpan dua Pepohon Kuad untuk dua imej, A dan B.

```
unsigned int A[] = {
    2,
    2, 2, 1, 0, 1, 0, 0, 2, 1, 0, 1, 0, 0,
    2, 0, 2, 0, 1, 0, 1, 0, 2, 0, 1, 0, 1,
    2, 2, 1, 0, 1, 0, 0, 2, 1, 0, 1, 0, 0,
    2, 0, 2, 0, 1, 0, 1, 0, 2, 0, 1, 0, 1,
};

unsigned int B[] = {
    2,
    2, 2, 0, 0, 1, 0, 1, 2, 1, 0, 0, 0, 1,
    2, 1, 2, 0, 0, 0, 1, 1, 2, 0, 1, 0, 0,
    2, 0, 1, 0, 1,
    2, 1, 0, 1, 0,
};
```

Dengan merujuk kepada segmen program di atas, jawab soalan-soalan di bawah. Untuk soalan-soalan (b) – (e), gunakan helaian jawapan yang disediakan di “Appendix: Helaian Jawapan untuk Soalan 6”. Ceraikan jawapan anda daripada kertas peperiksaan ini dan lampirkan dengan booklet jawapan anda.

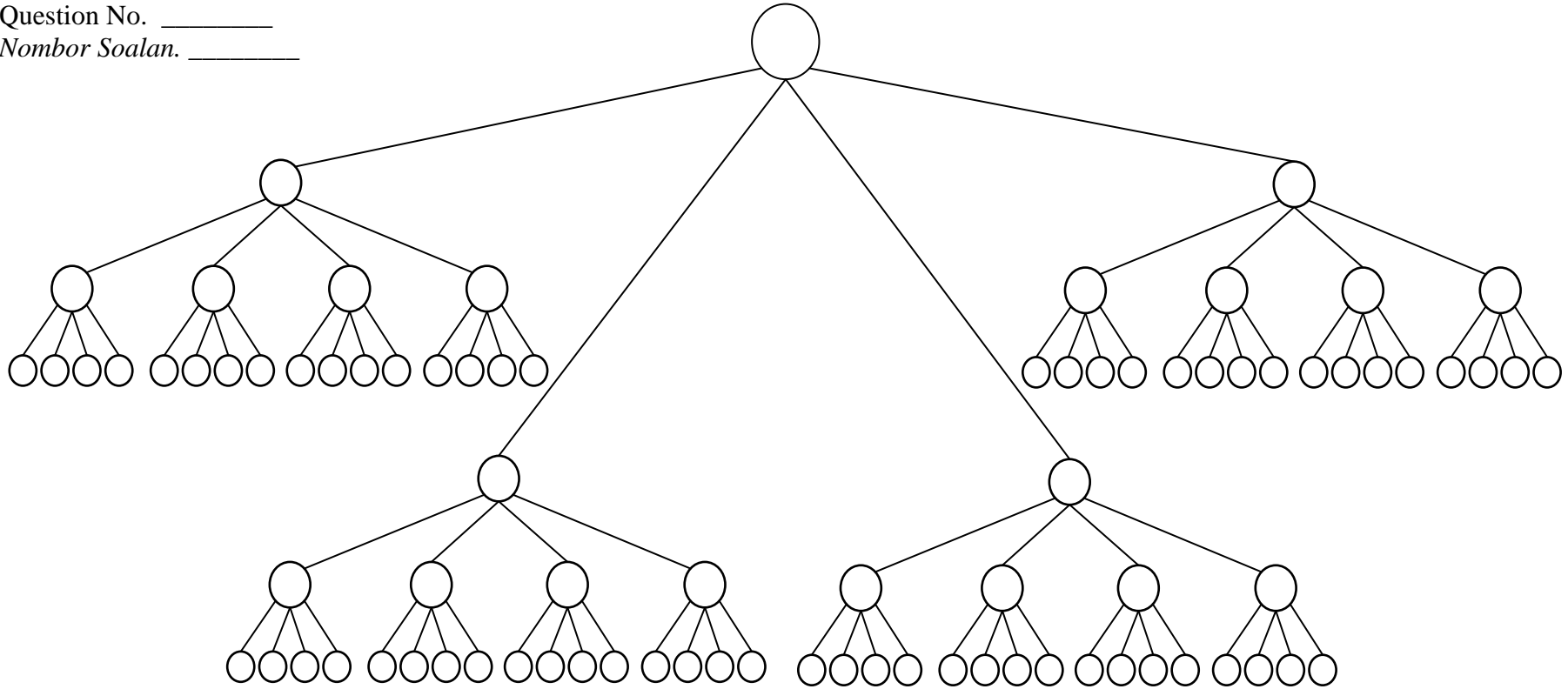
- (a) Tukarkan nilai-nilai tatasusunan-tatasusunan A dan B ke kod-kod perduaan yang mempunyai dua bit.
- (b) Dari kod-kod perduaan di (a), lukiskan pepohon-pepohon kuad untuk A dan B. Labelkan nod-nod pepohon dengan kod-kod perduaan.
- (c) Dari pepohon-pepohon kuad di (b), lorekkan perwakilan-perwakilan piksel untuk A dan B masing-masing.
- (d) Dari (c), cari perwakilan-perwakilan piksel untuk  $A \cdot B$  dan  $A \oplus B$ .
- (e) Dari (d), cari pepohon-pepohon kuad untuk  $A \cdot B$  dan  $A \oplus B$ . Labelkan nod-nod pepohon dengan kod-kod perduaan.
- (f) Dari (d) atau (e), cari perwakilan-perwakilan tatasusunan C++ untuk  $A \cdot B$  dan  $A \oplus B$ , sehingga peringkat 3, i.e. seperti dalam segmen program di atas untuk imej-imej A dan B.

[100 markah]

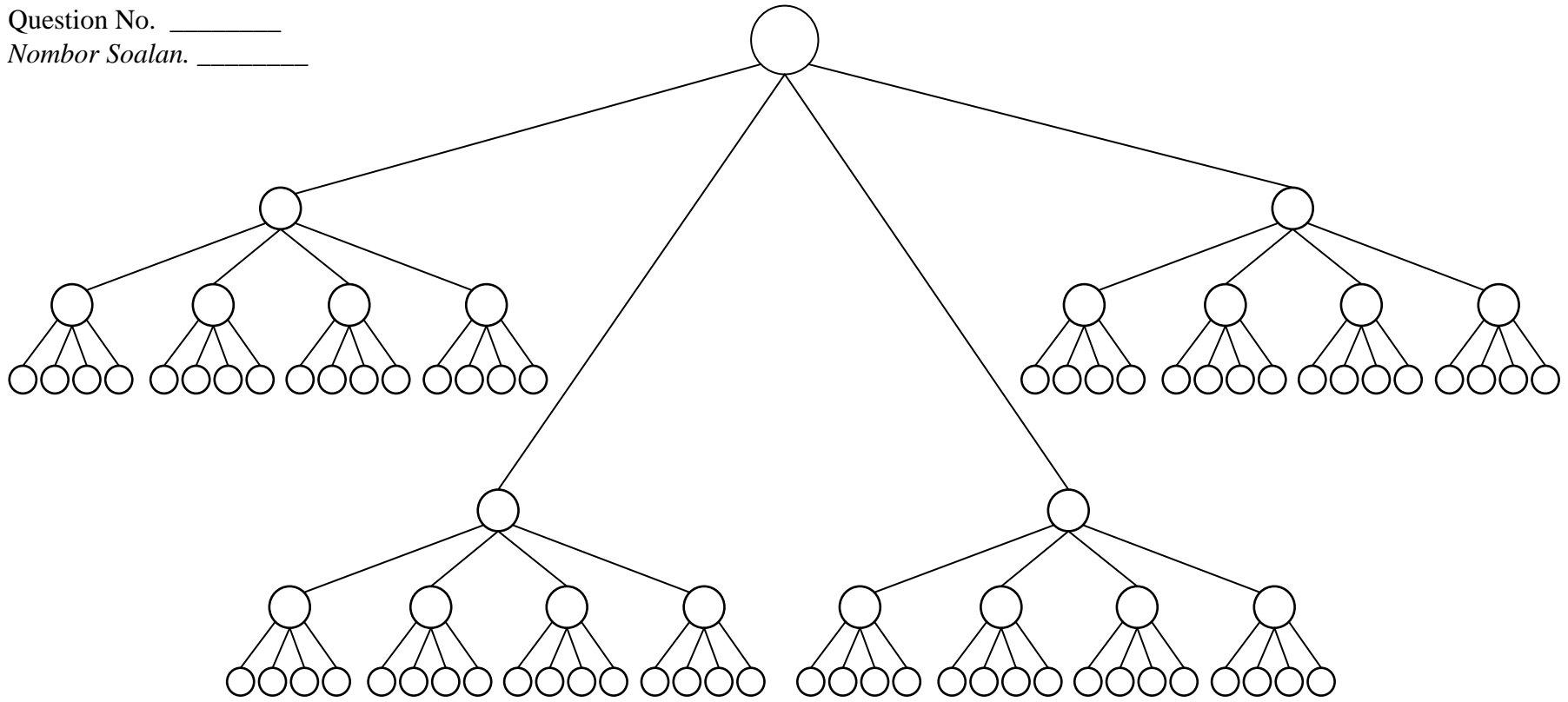




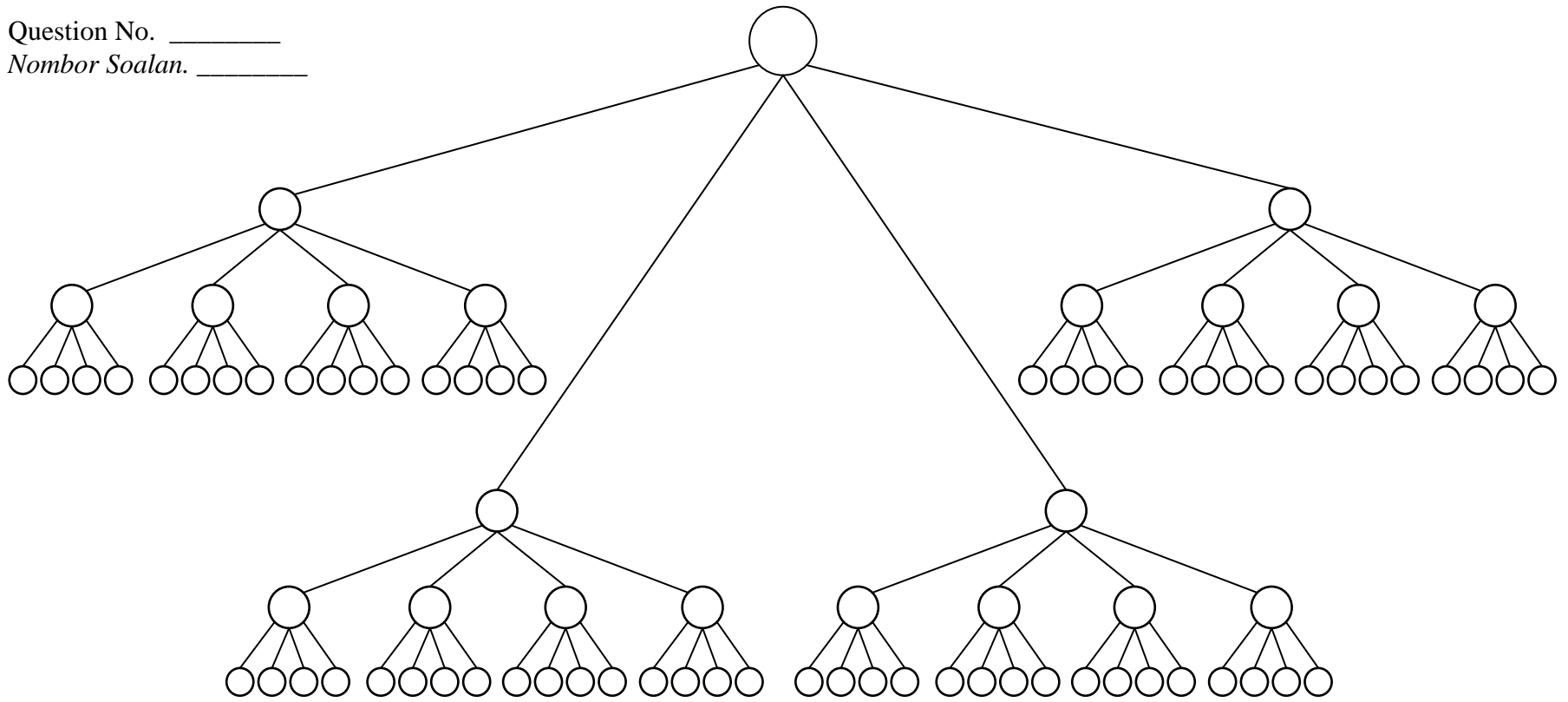
Question No. \_\_\_\_\_  
Nombor Soalan. \_\_\_\_\_



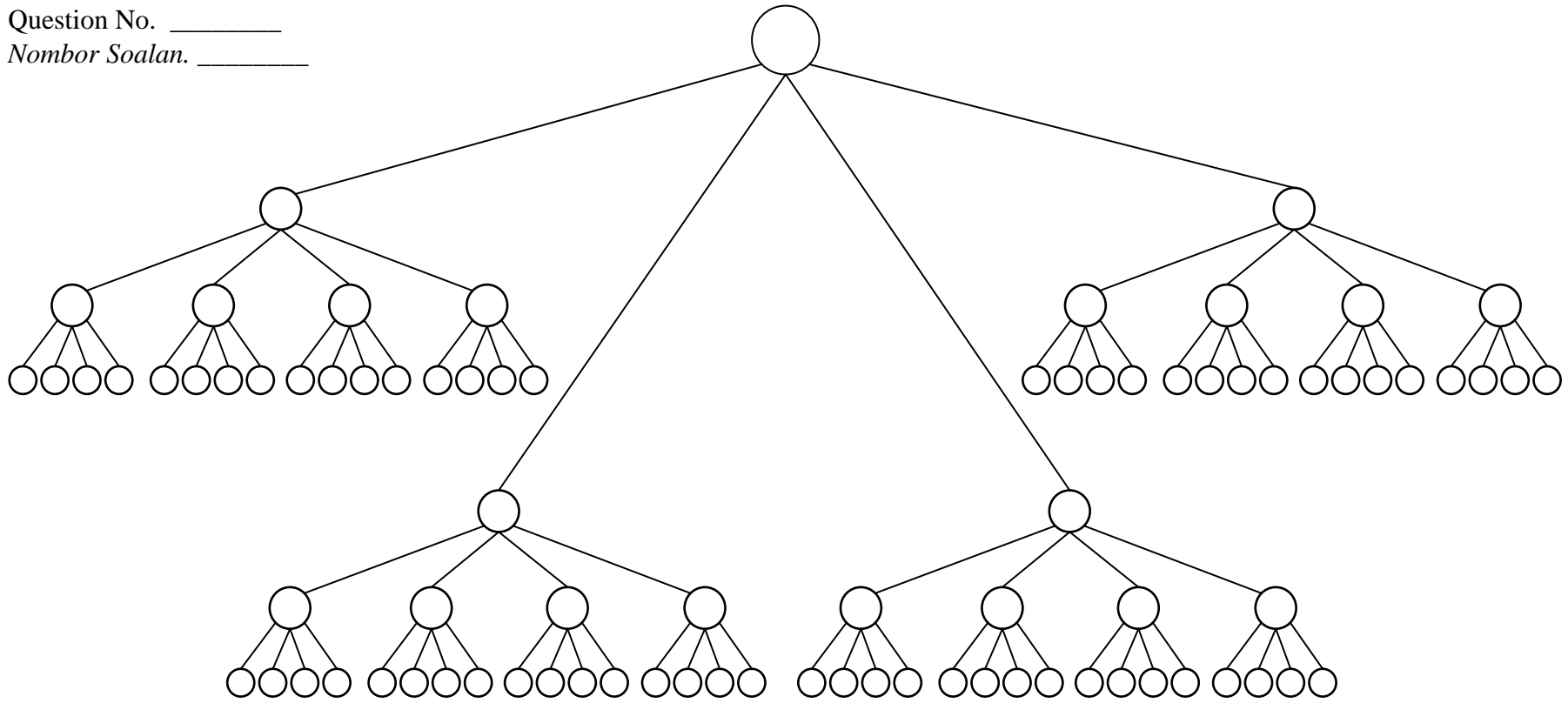
Question No. \_\_\_\_\_  
Nombor Soalan. \_\_\_\_\_



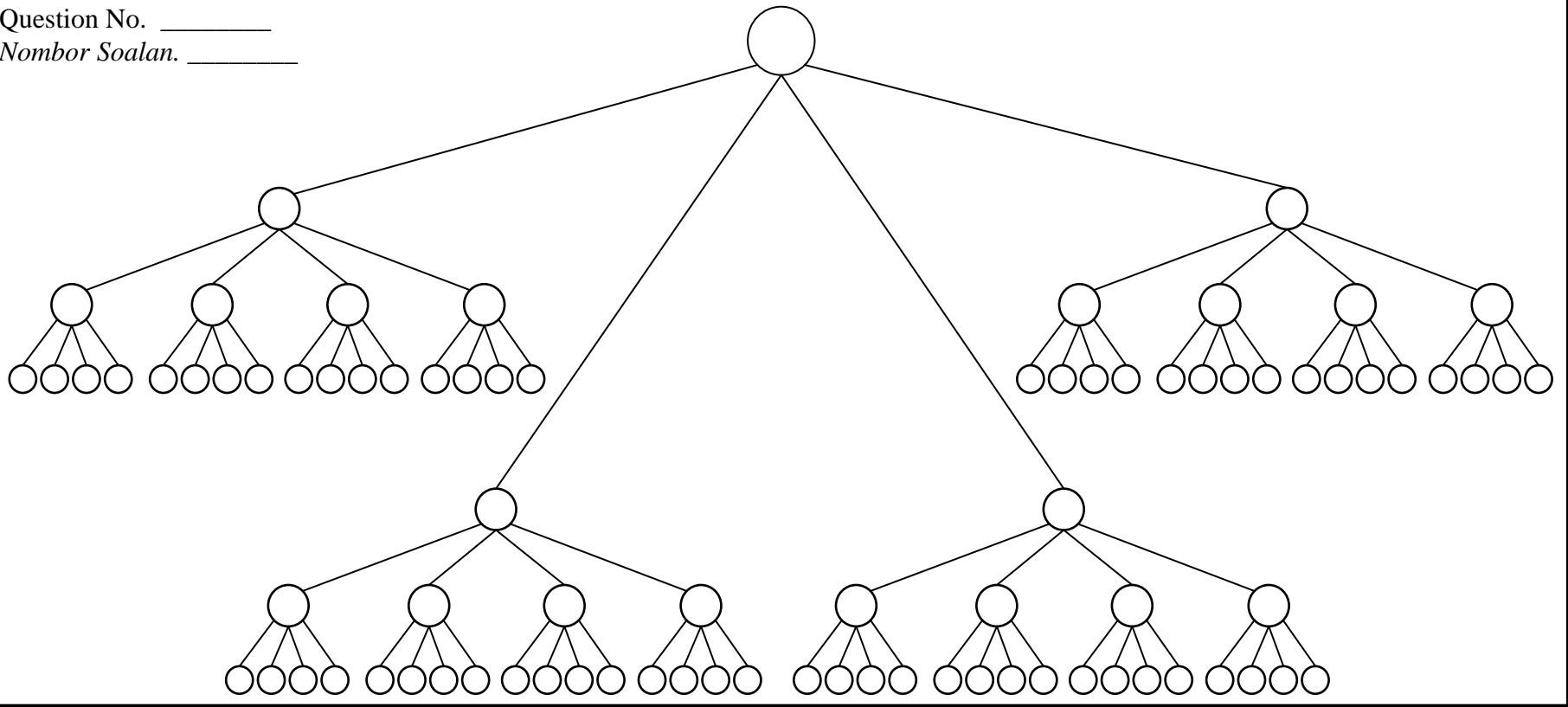
Question No. \_\_\_\_\_  
Nombor Soalan. \_\_\_\_\_



Question No. \_\_\_\_\_  
Nombor Soalan. \_\_\_\_\_

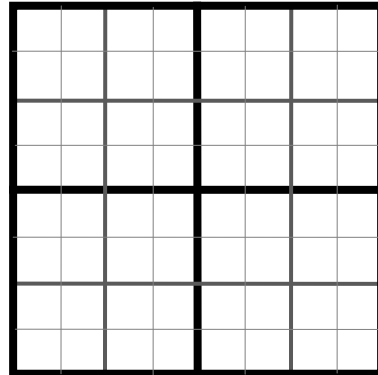
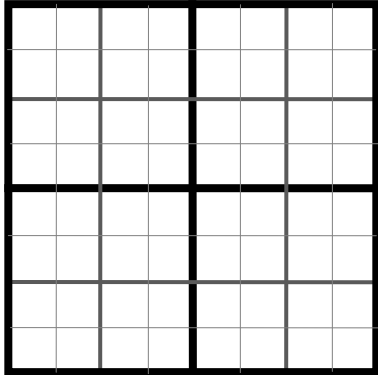


Question No. \_\_\_\_\_  
Nombor Soalan. \_\_\_\_\_



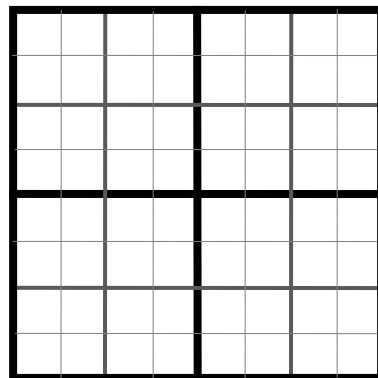
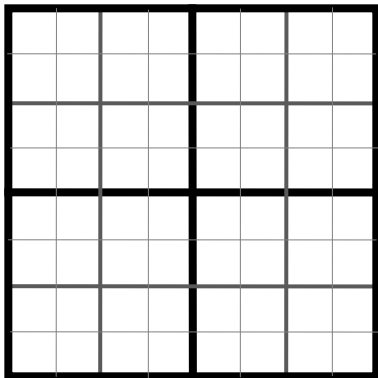
Question No. \_\_\_\_\_

*Nombor Soalan.* \_\_\_\_\_



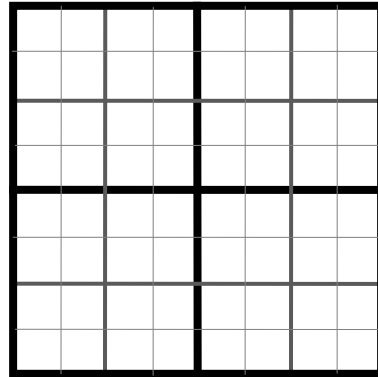
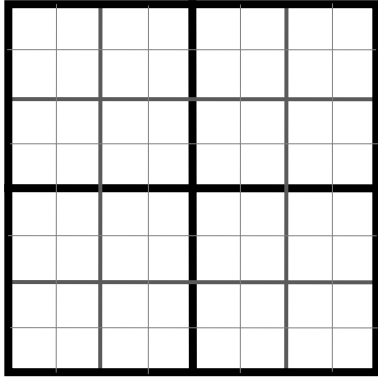
Question No. \_\_\_\_\_

*Nombor Soalan.* \_\_\_\_\_



Question No. \_\_\_\_\_

*Nombor Soalan.* \_\_\_\_\_



Question No. \_\_\_\_\_

*Nombor Soalan.* \_\_\_\_\_

