



UNIVERSITI TUN HUSSEIN ONN MALAYSIA

PEPERIKSAAN AKHIR SEMESTER II SESI 2008/2009

NAMA MATA PELAJARAN	:	STRUKTUR DATA
KOD MATA PELAJARAN	:	BIT 1073
KURSUS	:	1 BIT
TARIKH PEPERIKSAAN	:	APRIL/ MEI 2009
JANGKA MASA	:	2 JAM 30 MINIT
ARAHAN	:	JAWAB SEMUA SOALAN.DI BAHAGIAN A, DUA (2) SOALAN DI BAHAGIAN B DAN SATU (1) SOALAN DI BAHAGIAN C

KERTAS SOALAN INI MENGANDUNGI TIGA BELAS (13) MUKA SURAT

SECTION A

Instruction: Answer ALL questions.

Q1 Refer to the array statement below:

```
char country[8] = {'M', 'A', 'L', 'A', 'Y', 'S', 'I', 'A'};
```

- (a) What is the index of 'S'? (1 mark)
- (b) Write a statement in C language to refer to element 'Y' in the array. (1 mark)
- (c) What is the index for the last data in the array? (1 mark)
- (d) Write a statement in C language to print all the data in the array. (2 marks)

Q2 Draw a Binary Search Tree that fulfills the following traversal conditions:

Preorder : K D A B H P M S

Inorder : A B D H K M P S

Postorder : B A H D M S P K

(5 marks)

Q3 Given the following integer list:

80 8 60 110 34 2 500

Show a trace for each execution of:

- (a) Insertion sort. (5 marks)
- (b) Bubble sort. (5 marks)
- (c) Quick sort. (5 marks)

Q4 Given a list {26, 14, 8, 22, 37, 6, 44, 24, 53, 24, 10, 32},

- (a) Draw a Binary Search Tree. (3 marks)
- (b) Based on the answer in **Q4 (a)**, write the sequence of nodes when you traverse the tree using
- (i) inorder,
 - (ii) preorder and
 - (iii) postorder methods. (6 marks)
- (c) Draw a new Binary Search Tree after inserting 15 and 40 in the list. (3 marks)
- (d) Draw a new Binary Search Tree after deleting 22 and 37 from the list. (3 marks)

Q5 Consider this snippet of code:

```
int find ( int a[], int n, int x )
{
    int i;
    for ( i = 0; i < n; i++ ) {
        if ( a[i] == x )
            return i;
    }
    return 0;
}
```

- (a) What is the complexity of this algorithm? (3 marks)
- (b) What is this search called? (2 marks)

SECTION B

Instruction: Answer **TWO (2)** from **THREE (3)** questions only.

Q6 Given **Figure Q6**, answer the questions below:

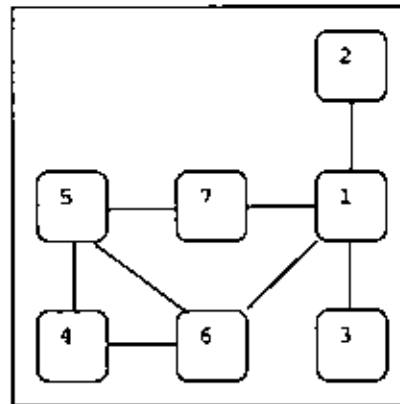


Figure Q6

- (a) List all unique cycles in **Figure Q6** above. (3 marks)
- (b) Generate output of the graph in **Preorder** traversal using vertex '1' as a starting point. (3 marks)
- (c) Generate output of the graph in **Postorder** traversal using vertex '1' as a starting point. (4 marks)

Q7 Given:

$$\text{Catalan}(0) = 1$$

$$\text{Catalan}(n) = \text{Catalan}(n-1) * (4n+2) / (n+1), \text{ for } n > 0$$

Write a recursive function that calculates $\text{Catalan}(n)$. Make sure that your code considers all integer arguments.

(10 marks)

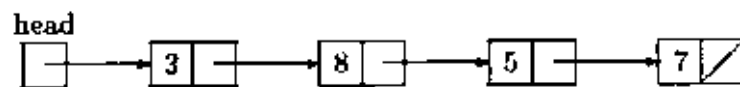
Q8 For these questions, use the following definition for a Node:

```
struct Node {
    int item;
    Node next;
}
```

(a) Consider the following function that has access to the definition of Node:

```
void someFunc(Node head)
{
    Node p = head;
    Node q;
    while (p != null && p.next != null)
    {
        q = p.next;
        p.item = p.item + q.item;
        p.next = q.next;
        p = p.next;
    }
}
```

Show the end result of executing `someFunc(head)` for the following list:



(4 marks)

(b) Using the above definition of Node, write a function `countOddPairs(Node head)` which returns a count of the number of pairs of adjacent nodes in the list whose sum is odd. For example, calling `countOddPairs(head)` for the above list should return 2, since $3 + 8$ and $8 + 5$ are both odd, while $5 + 7$ is even.

(6 marks)

SECTION C

Instruction: Answer **ONE (1)** from **TWO (2)** questions only.

Q9 Here is C code for one version of the xSort algorithm:

```
void xSort(int [ ] a, int n)
{
    for (int k = 1; k < n; k++) {
        for (int i = k; i > 0 && a[i - 1] > a[i]; i--)
        {
            int temp = a[i - 1];
            a[i - 1] = a[i];
            a[i] = temp;
        }
    }
}
```

- (a) Trace the operation of xSort by showing the contents of a() at the end of each pass through the outer loop; use the array {5, 1, 2, 3, 4}, where n = 5:

k	a[0]	a[1]	a[2]	a[3]	a[4]
initial	5	1	2	3	4
1					
2					
3					
4					

(4 marks)

- (b) How many item comparisons ($a[i - 1] > a[i]$) are performed in **Q9(a)**?
(3 marks)
- (c) How many swaps are performed in part **Q9(a)**?
(3 marks)
- (d) What would be the numbers of comparisons and swaps for the array {50, 1, 2, ..., 49}, where n = 50?
(5 marks)

Q10 Here is C code for one version of the ySort algorithm:

```
void ySort(int [ ] a, int n)
{
    for (int k = n - 1; k > 0; k--) {
        int largest = 0;
        for (int i = 1; i <= k; i++) {
            if (a[i] > a[largest]) largest = i;
        }
        // move largest item to position k
        int temp = a[largest];
        a[largest] = a[k];
        a[k] = temp;
    }
}
```

- (a) Trace the operation of ySort by showing the contents of a[] at the end of each pass through the outer loop; use the array {5, 1, 2, 3, 4}, where n = 5:

k	a[0]	a[1]	a[2]	a[3]	a[4]
initial	5	1	2	3	4
4					
3					
2					
1					

(4 marks)

- (b) How many item comparisons (a[i] > a[largest]) are performed in part Q10(a)?

(3 marks)

- (c) How many swaps are performed in part Q10(a)?

(3 marks)

- (d) What would be the numbers of comparisons and swaps for the array {50, 1, 2, ..., 49}, where n = 50?

(5 marks)

BAHAGIAN A

Arahan: Jawab **SEMUA** soalan.

S1 Diberi pernyataan tatasusunan berikut:

```
char negara[8] = {'M', 'A', 'L', 'A', 'Y', 'S', 'I', 'A'};
```

- (a) Apakah indeks bagi 'S'? (1 markah)
- (b) Tuliskan pernyataan dalam bahasa C yang merujuk kepada 'Y' di dalam tatasusunan. (1 markah)
- (c) Apakah indeks bagi nilai yang terakhir dalam tatasusunan? (1 markah)
- (d) Tuliskan pernyataan dalam bahasa C untuk memaparkan semua nilai dalam tatasusunan. (2 markah)

S2 Lukis Pepohon Carian Binari yang memenuhi syarat aturan berikut:

Pra-tertib : K D A B H P M S

Mengikut tertib : A B D H K M P S

Pasca-tertib : B A H D M S P K

(5 markah)

S3 Berdasarkan susunan senarai integer di bawah:

80 8 60 110 34 2 500

Tunjukkan jejak bagi setiap operasi:

- (a) Isihan Selitan. (5 markah)
- (b) Isihan Buih. (5 markah)

- (c) Isihan Pantas. (5 markah)

S4 Berdasarkan senarai {26, 14, 8, 22, 37, 6, 44, 24, 53, 24, 10, 32},

- (a) Lukiskan Pepohon Carian Binari. (3 markah)
- (b) Berdasarkan jawapan S4(a), tuliskan aturan nod-nod yang dilawati mengikut tertib, pra-tertib dan pasca-tertib. (6 markah)
- (c) Lukiskan semula pepohon carian binari apabila 15 dan 40 diselitkan. (3 markah)
- (d) Lukiskan semula pepohon carian binari apabila 22 dan 37 dihapuskan. (3 markah)

S5 Diberi keratan kod di bawah:

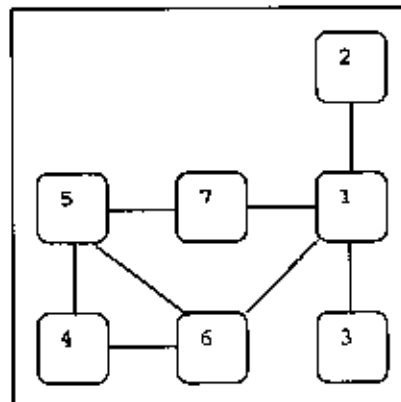
```
int find ( int a[], int n, int x )
{
    int i;
    for ( i = 0; i < n; i++ ) {
        if ( a[i] == x )
            return i;
    }
    return 0;
}
```

- (a) Apakah kompleksiti algoritma ini? (3 markah)
- (b) Apakah nama bagi carian tersebut? (2 markah)

BAHAGIAN B

Arahan: Jawab **DUA (2)** daripada **TIGA (3)** soalan sahaja.

S6 Diberi **Rajah S6**, jawab soalan di bawah.



Rajah S6

- (a) Senaraikan semua kitaran unik dalam **Rajah S6** di atas. (3 markah)
- (b) Jana *output* bagi graf dalam aturan **pra-tertib** dengan menggunakan verteks '1' sebagai titik permulaan. (3 markah)
- (c) Jana *output* bagi graf dalam aturan **pasca-tertib** dengan menggunakan verteks '1' sebagai titik permulaan. (4 markah)

S7 Diberikan:

$$\text{Catalan}(0) = 1$$

$$\text{Catalan}(n) = \text{Catalan}(n-1) * (4n+2) / (n+1), \text{ for } n > 0$$

Tuliskan satu fungsi **rekursif** yang mengira Catalan(n). Pastikan kod anda mengambilkira kesemua nilai integer.

(10 markah)

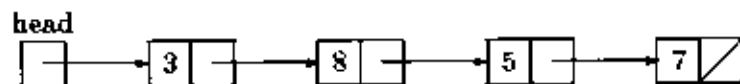
S8 Bagi soalan berikut, gunakan pengistiharan Node dibawah:

```
struct Node {
    int item;
    Node next;
}
```

(a) Fungsi dibawah mempunyai akses kepada pengistiharan Node di atas:

```
void someFunc(Node head)
{
    Node p = head;
    Node q;
    while (p != null && p.next != null)
    {
        q = p.next;
        p.item = p.item + q.item;
        p.next = q.next;
        p = p.next;
    }
}
```

Berikan keputusan akhir setelah pelaksanaan `someFunc(head)` untuk senarai dibawah:



(4 markah)

(b) Dengan pengistiharan node di atas, tulis satu fungsi `countOddPairs(Node head)` yang mengembalikan bilangan pasangan *node* dimana jumlah nilainya ganjil. Sebagai contoh, memanggil `countOddPairs(head)` bagi *list* di atas akan mengembalikan nilai 2, kerana $3 + 8$ and $8 + 5$ adalah ganjil, tetapi $5 + 7$ adalah genap.

(6 markah)

BAHAGIAN C

Arahan: Jawab SATU (1) daripada DUA (2) soalan sahaja.

S9 Diberikan kod C untuk algoritma xSort:

```
void xSort(int[] a, int n)
{
    for (int k = 1; k < n; k++) {
        for (int i = k; i > 0 && a[i - 1] > a[i]; i--)
        {
            int temp = a[i - 1];
            a[i - 1] = a[i];
            a[i] = temp;
        }
    }
}
```

- (a) Jejak operasi xSort dengan menunjukkan kandungan a() pada penghujung setiap laluan melalui gelung luaran; gunakan tatasusunan {5, 1, 2, 3, 4}, dimana n = 5:

k	a[0]	a[1]	a[2]	a[3]	a[4]
mula	5	1	2	3	4
4					
3					
2					
1					

(4 markah)

- (b) Berapa kalikah perbandingan item ($a[i - 1] > a[i]$) dilaksanakan di Bahagian S9(a)?

(3 markah)

- (c) Berapa kalikah tukaran (*swap*) dilaksanakan di Bahagian S9(a)?

(3 markah)

- (d) Berapa kalikah perbandingan dan tukaran akan dilakukan bagi tatasusunan {50, 1, 2, ..., 49}, di mana n = 50?

(5 markah)

S10 Diberikan kod C untuk satu versi algoritma ySort:

```
void ySort(int [ ] a, int n)
{
    for (int k = n - 1; k > 0; k--) {
        int largest = 0;
        for (int i = 1; i <= k; i++) {
            if (a[i] > a[largest]) largest = i;
        }
        // move largest item to position k
        int temp = a[largest];
        a[largest] = a[k];
        a[k] = temp;
    }
}
```

- (a) Jejak operasi ySort dengan menunjukkan kandungan a[] pada penghujung setiap laluan melalui gelang luaran; gunakan tatasusunan {5, 1, 2, 3, 4}, di mana n = 5:

k	a[0]	a[1]	a[2]	a[3]	a[4]
mula	5	1	2	3	4
4					
3					
2					
1					

(4 markah)

- (b) Berapa kalikah perbandingan item ($a[i] > a[largest]$) dilaksanakan di Bahagian S10(a)?

(3 markah)

- (c) Berapa kalikah tukaran (*swap*) dilaksanakan di Bahagian S10(a)?

(3 markah)

- (d) Berapa kalikah perbandingan dan tukaran akan dilakukan bagi tatasusunan {50, 1, 2, ..., 49}, di mana n = 50?

(5 markah)