

108-1 Data Structure Midterm

1. (1) & % A+B / C D== >=E F + + * G H % P M K //有一點錯就沒分
(2) (a) 3 //a~c 不論寫算式還是結果, 都有分
(b) 8/2-3
(c) 1
(3) 9 //4分

2. (1) //錯一個扣一分 3分

5	4	7
0	0	1
1	1	4
2	2	2
3	0	5
3	3	9
4	1	8
4	3	3

- (2) //錯一個扣一分 3分

4	5	7
0	0	1
0	3	5
1	1	4
1	4	8
2	2	2
3	3	9
3	4	3

- (3) //錯一個扣一分 3分

5	3	8
0	0	7
0	1	5
1	2	4
2	1	8
3	0	35
3	1	25
3	2	27
4	2	17

3. (1) 7^n //2 分
 (2) 10^n //2 分
 (3) $n \log n$ //2 分

4. (1) //4 分

```

newnode->llink = node; (a)
newnode->rlink = node->rlink; (b)
node->rlink->llink = newnode; (c)
node->rlink = newnode; (d)
  
```

(2) //4 分

```

deleted->llink->rlink= deleted->rlink; (a)
deleted->rlink->llink= deleted->llink; (b)
free(deleted);
  
```

5. //6 分

	space	time
2D array	$O(\text{rows} * \text{cols})$	$O(\text{rows} * \text{cols})$
Transpose	$O(\text{elements})$	$O(\text{cols} * \text{elements})$
Fast Transpose	$O(\text{elements} + \text{MAX_COL})$	$O(\text{cols} + \text{elements})$

6. 3 11 7 11 //4 分, 錯一個扣一分

7. (1)(2) //1分 //1分

■ Definition

An *algorithm* is a finite set of instructions that accomplishes a particular task.

■ Criteria

- input
- output
- definiteness: clear and unambiguous
- finiteness: terminate after a finite number of steps
- effectiveness: instruction is basic enough to be carried out

CHAPTER 1

8

(3) //1分

■ Abstract Data Type (ADT)

An *ADT* is a data type that is organized in such a way that **the specification of the objects and the operations on the objects** is separated from

- the representation of the objects .
- the implementation of the operations.

8. (1) 0066FF18 //1分

(2) 0066FF18 //1分

(3) 0066FF18 //1分

(4) 0061FF1C //1分

(5) 10 //1分

(6) 10 //送分

9. (1) $S[i] + y$ //3分

(2) y //3分

(3) $S[i] + y$ //4分

10.

(1) $k+1$ //5分

(2) n //5分

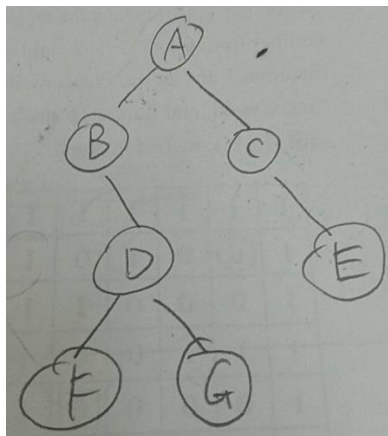
11.

//10分 共有16步分

4. 44 43 42 41
5. 44 43 42
6. 44 43 42 33
7. 44 43 42 33 22
8. 44 43 42 33 22 21
9. 44 43 42 33 22 21 12
10. 44 43 42 33 22 21 12 23
11. 44 43 42 33 22 21 12 23 14
12. 44 43 42 33 22 21 12 23
13. 44 43 42 33 22 21 12 23 34
14. 44 43 42 33 22 21 12 23
15. 44 43 42 33 22 21 12
16. 44 43 42 33 22 21 12 11

12.(1)

//2分



(2) (a)2 //2分

(b)4 //2分

(3) 3 //2分

13. (1)16 //2分

(2) 1023 //2分

(3) 66 //3分

(4) 9 //3分