

[DB 8강]-자료 구조(선형, 비선형구조) [출제빈도 '상']

1. 자료구조 분류

: 자료를 기억장치 내에 저장하는 방법

- 1) 선형 구조 : 순차 리스트 (스택, 큐, 데크, 배열), 연결 리스트
- 2) 비선형 구조 : Tree, Graph

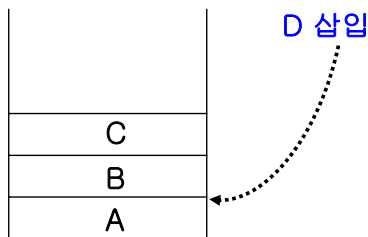
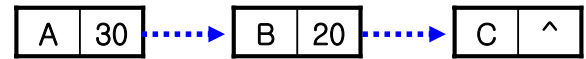
2. 선형 구조 ★☆☆☆☆

1) 순차 리스트 (선형, Sequential List) : 연속적인 기억장소에 저장

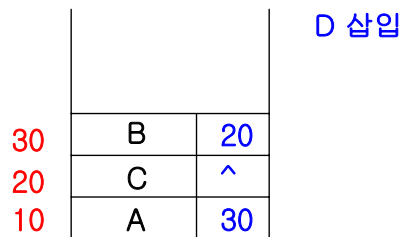
- 아파트를 계단으로 연속적으로 이동
- 특징 : 구조 간단, 기억장소 이용 효율이 높음, 삽입/삭제 어려움, 연결리스트에 비해 검색 빠름.

2) 연결 리스트 (Linked List) : 비연속적으로 저장

- 아파트를 엘리베이터로 비연속적으로 이동 (포인터)
- 특징 : 기억장소 이용 효율이 낮음, 삽입/삭제 용이, 순차 리스트에 비해 검색 느림.



1 [순차 리스트]

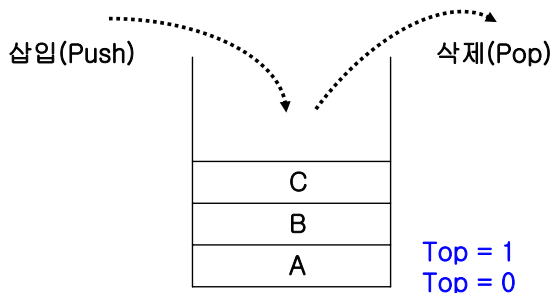


[연결 리스트]

[DB 8강]-자료 구조(선형, 비선형구조)

3. 선형구조 > 순차리스트 > 스택 (Stack) ★★★★★

1) 삽입/삭제가 한 쪽에서 이루어지는 데이터 구조 (LIFO : Last In First Out)



- * Top Point : 가장 최근에 삽입된 자료 또는 가장 먼저 삭제될 자료를 가리키는 스택 포인터
- 삽입 : Top 값 증가
- 삭제 : Top 값 감소

2) 스택 응용 분야 ★

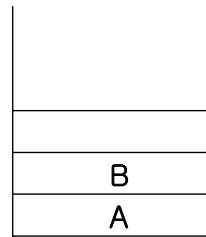
- 인터럽트의 처리
- 수식의 계산
- 서브루틴의 복귀번지 저장
- 부프로그램(sub program)의 호출 = 함수 호출의 순서제어

운영체제의 작업 스케줄링(X)

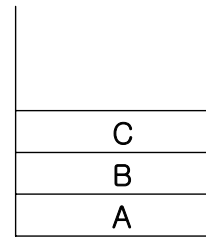
[DB 8강]-자료 구조(선형, 비선형구조)

2) 삽입 알고리즘

```
Top = Top + 1
If(Top > M) Then
    Stack_overflow
Else
    Stack(Top) ← data
```



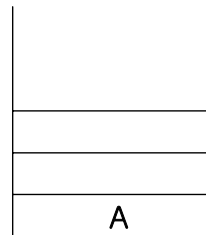
Top = 2 , M = 3



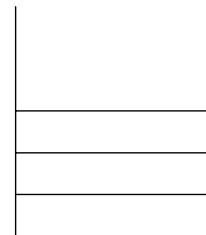
Top = 3 , M = 3

3) 삭제 알고리즘

```
If(Top = 0) Then
    Stack_Empty
Else
    data ← Stack(Top)
    Top = Top - 1
```



Top = 1



Top = 0

[DB 8강]-자료 구조(선형, 비선형구조)

스택에 데이터를 A, B, C, D 순으로 저장했을 경우, 이들 데이터가 출력되는 결과로 가능한 것은 ? [정답 나]

- 가. D-B-C-A 나. C-B-D-A
다. D-C-A-B 라. D-A-C-B

=> 문제에서 저장했을 경우를 저장할 경우로 해석해야 이 문제를 해결할 수 있습니다.

가 : A(PUSH) -> B(PUSH) -> C(PUSH) -> D(PUSH) -> D(POP)
-> 그 다음 B(POP) 할 수 없습니다. C(POP)만 가능하죠. C보다 B를 POP 할 수 없습니다.

나 : A(PUSH) -> B(PUSH) -> C(PUSH) -> C(POP) -> B(POP) -> D(PUSH) -> D(POP) -> A(POP)
-> 이해하실 내용은 꼭 A,B,C,D를 다 PUSH 후에 POP을 할 필요 없다는 것입니다.

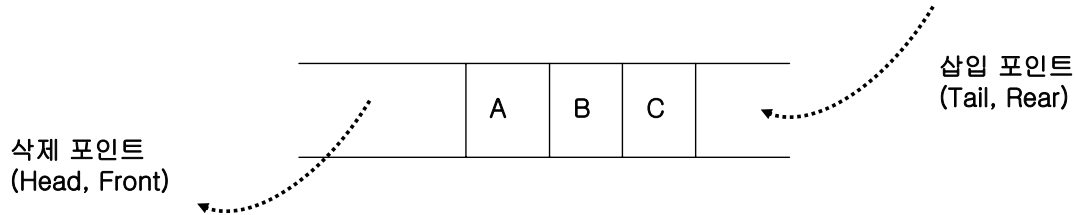
다 : A(PUSH) -> B(PUSH) -> C(PUSH) -> D(PUSH) -> D(POP) -> C(POP)
-> 그 다음 A(POP) 할 수 없습니다.

라 : A(PUSH) -> B(PUSH) -> C(PUSH) -> D(PUSH) -> D(POP)
-> 그 다음 A(POP) 할 수 없습니다.

[DB 8강]-자료 구조(선형, 비선형구조)

4. 선형구조 > 순차리스트 > 큐(Queue) ★★★★★

1) 노드의 삽입 작업은 선형 리스트의 한 쪽 끝에서, 제거 작업은 다른 쪽 끝에서 수행되는 자료 구조 (FIFO : First In First Out)

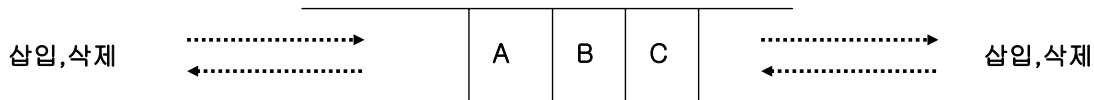


2) 응용 분야

- 운영체제의 작업 스케줄링, 키보드 버퍼 이용 시, 스푼(spool) 운용 시

5. 선형구조 > 순차리스트 > 덱(Deque) ★★★★★

1) 삽입과 삭제가 리스트의 양쪽 끝에서 발생할 수 있는 자료 구조 (double ended queue)



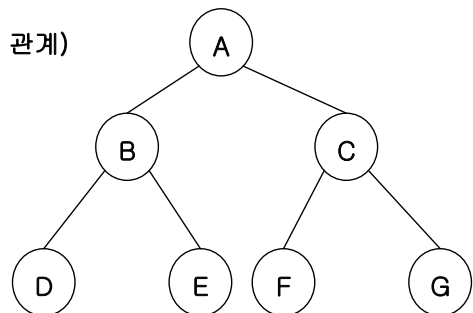
[DB 8강]-자료 구조(선형, 비선형구조)

1. 비선형구조 > 트리 (Tree) ★★★★★

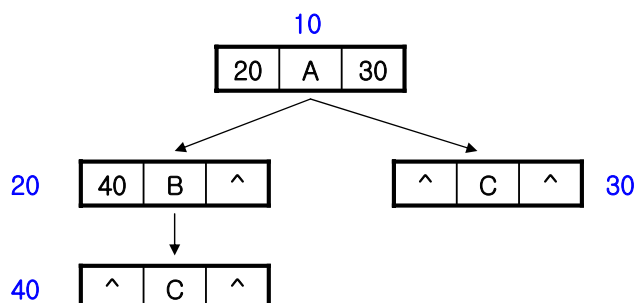
: 노드와 간선으로 구성되어 있고, 사이클이 없다. (족보, 부모 자식 관계)

1) 용어

- Node (노드)
- Root Node (근노드) : 최상위 노드
- 노드의 차수 (degree) : 어떤 노드의 서브트리 수
- 트리의 차수 : 노드의 차수 값 중 최대값
- 단말노드 (Terminal, Leaf) : 차수가 0 인 노드
- 형제노드 : 같은 부모 노드를 가지는 노드
- 레벨(Level) : 노드의 깊이



2) 특징 : 연결 리스트 구조로 표현 (포인터 이용)



[DB 8강]-자료 구조(선형, 비선형구조)

2. 트리의 종류 > 이진 트리 ★☆☆☆☆

: 차수가 2 이하로 구성된 트리

- 이진 트리의 레벨 i 에서 최대 노드의 수 : $2^i - 1$

3. 트리의 종류 > 스레드(threaded) 이진 트리 ★☆☆☆☆

: 기억 공간의 낭비 원인이 되는 널 링크 부분을 트리 순회 시 이용되도록 구성한 트리

- 널 링크를 다른 노드를 가리키는 포인터로 대체한다.
- 스택의 도움 없이 트리를 순회할 수 있는 장점이 있다.
- 실제 포인터와 스레드를 구별하기가 어렵다

[DB 8강]-자료 구조(선형, 비선형구조)

4. 트리의 종류 > B - 트리 ★☆☆☆☆

: 인덱스 파일에서 인덱스를 구성하는 방법 중의 하나

- 한 노드 안에 있는 키값은 오름차순을 유지한다.
- 모든 리프노드(단노드)는 같은 레벨에 있다. (균형 유지)
- 루트노드는 리프가 아닌 이상 적어도 두개의 서브트리를 갖는다.
- 탐색, 추가, 삭제는 루트로부터 시작한다.
- 인덱스 파일에서 인덱스를 구성하는 방법 중 하나다.

100

200	30	300	60	400
-----	----	-----	----	-----

200

	10		20	
--	----	--	----	--

300

	40		50	
--	----	--	----	--

400

	70		80	
--	----	--	----	--

5	8	15	19	25	29	35	36	45	46	52	53	65	66	75	76	81	82
---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

[DB 8강]-자료 구조(선형, 비선형구조)

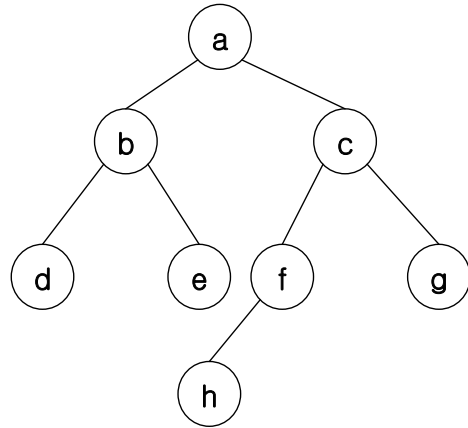
5. 이진 트리 운행법 ★★★★★☆

: 트리를 구성하는 노드들을 찾아가는 방법

- Preorder (전위) : Root → Left → Right
- Inorder (중위) : Left → Root → Right
- Postorder (후위) : Left → Right → Root

예)

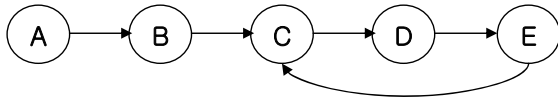
- Preorder (전위) : a, b, d, e, c, f, h, g
- Inorder (중위) : d, b, e, a, h, f, c, g
- Postorder (후위) : d, e, b, h, f, g, c, a



6. Graph ★★☆☆☆☆

: 노드와 간선으로 구성되어 있고, 사이클이 있다.

1) 그래프를 인접행렬로 표시



	A	B	C	D	E
A	0	1	0	0	0
B	0	0	1	0	0
C	0	0	0	1	0
D	0	0	0	0	1
E	0	0	1	0	0

9

[DB 8강]-자료 구조(선형, 비선형구조)

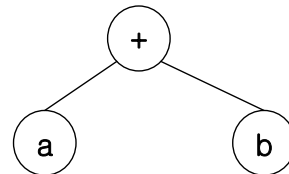
7. 수식 표기법 변환 ★★★★★☆

: 산술식을 계산하기 위해 기억공간에 기억시키는 방법으로 이진 트리를 많이 이용한다.

- PreFix (전위 표기법) : 연산자 → Left 피연산자 → Right 피연산자
- InFix (중위 표기법) : Left 피연산자 → 연산자 → Right 피연산자
- PostFix (후위 표기법) : Left 피연산자 → Right 피연산자 → 연산자

예)

- PreFix (전위 표기법) : + a b
- InFix (중위 표기법) : a + b
- PostFix (후위 표기법) : a b +



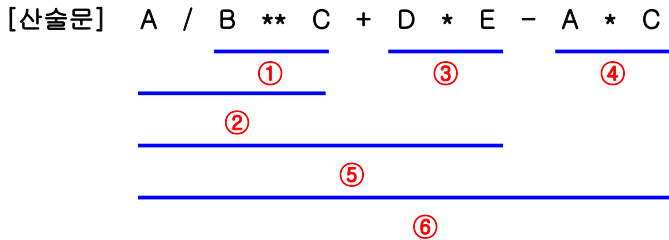
10

[DB 8강]-자료 구조(선형, 비선형구조)

주의)

- 연산 우선 순위 : 괄호 -> ** (거듭제곱) -> *, / -> +, -

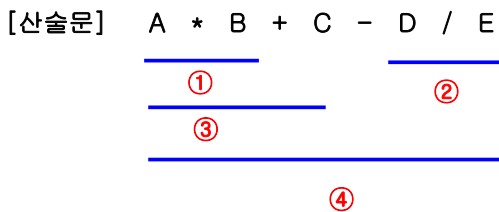
1) InFix -> PostFix



[풀이]

- ① : $BC**$
- ② : $A / ① \rightarrow ABC** /$
- ③ : $DE*$
- ④ : $AC*$
- ⑤ : $② + ③ \rightarrow ABC** / DE** +$
- ⑥ : $⑤ - ④ \rightarrow ABC** / DE** + AC* -$

2) InFix -> PreFix



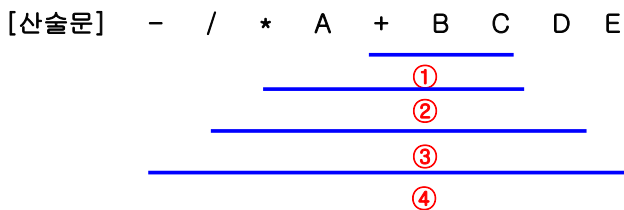
[풀이]

- ① : $*AB$
- ② : $/DE$
- ③ : $① + C \rightarrow ++ABC$
- ④ : $③ - ② \rightarrow -++ABC/DE$

[DB 8강]-자료 구조(선형, 비선형구조)

주의) : 왼쪽에서 PreFix 표기인 연산자, 피연산자, 피연산자 구조를 찾는다.

3) PreFix -> PostFix

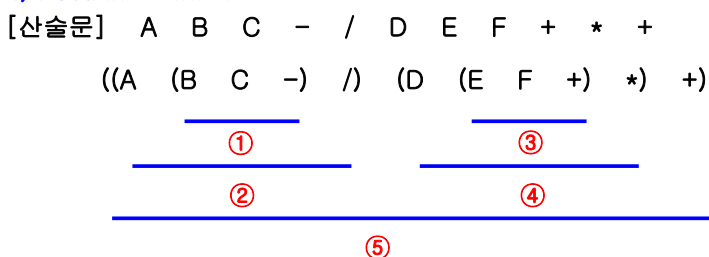


[풀이]

- ① : $BC+$
- ② : $* A ① \rightarrow ABC**$
- ③ : $/ ② D \rightarrow ABC**D /$
- ④ : $- ③ E \rightarrow ABC**D/E -$

주의) : 왼쪽에서 PostFix 표기인 피연산자, 피연산자, 연산자 구조를 찾아서 괄호로 묶는다.

4) PostFix -> InFix



[풀이]

- ① : $(B-C)$
 - ② : $(A ① /) \rightarrow (A/(B-C))$
 - ③ : $(E+F)$
 - ④ : $(D ③ *) \rightarrow (D*(E+F))$
 - ⑤ : $(② ④ +) \rightarrow ((A/(B-C))+(D*(E+F)))$
- > 필요없는 괄호 없애기
: $A/(B-C)+D*(E+F)$

[DB 8강]-자료 구조(선형, 비선형구조)

1. 순차 리스트(Sequential List)가 아닌 것은?

- 가. 배열(Array) 나. 트리(Tree)
다. 데크(Deque) 라. 스택(Stack)

2. 순차적인 선형 구조(Sequential Linear Structure)에 해당되는 자료 구조는?

- 가. 트리 나. 연결 리스트
다. 그래프 라. 큐

3. 비선형 구조와 선형 구조가 옳게 짝지어진 것은?

- ①스택(Stack) ②큐(Queue) ③트리(Tree)
④연결 리스트(Linked List) ⑤그래프(Graph)

- 가. 비선형 구조 : ①,②,⑤ 선형구조 : ③,④
나. 비선형 구조 : ③,⑤ 선형구조 : ①,②,④
다. 비선형 구조 : ①,②,③ 선형구조 : ④,⑤
라. 비선형 구조 : ③ 선형구조 : ①,②,④,⑤

[정답] 1.나 2.라 3.나 4.라 5.나 6.가

13

4. 선형 리스트의 특징이 아닌 것은?

- 가. 가장 간단한 데이터 구조 중 하나이다.
나. 배열과 같이 연속되는 기억장소에 저장되는 리스트를 말한다.
다. 기억장소 효율을 나타내는 메모리 밀도가 1이다.
라. 데이터 항목을 추가, 삭제하는 것이 용이하다.

5. 연결 리스트(Linked List)에 대한 설명으로 거리가 먼 것은?

- 가. 노드의 삽입이나 삭제가 쉽다.
나. 노드들이 포인터로 연결되어 검색이 빠르다.
다. 연결을 해주는 포인터(Pointer)를 위한 추가 공간이 필요하다.
라. 연결 리스트 중에는 중간 노드 연결이 끊겨지면 그 다음 노드를 찾기 힘들다.

6. 회소 행렬을 링크드 리스트(Linked List)로 표현할 때 가장 큰 장점은?

- 가. 기억장소가 절약된다.
나. 임의 위치 액세스(Random Access)가 가능하다.
다. 이진 검색(Binary Search)이 가능하다.
라. 행렬간의 연산시간을 줄일 수 있다.



[DB 8강]-자료 구조(선형, 비선형구조)

7. 리스트 내의 데이터 삽입, 삭제가 한 쪽 끝에서 이루어지는 데이터 구조를 무엇이라 하는가?

- 가. 스택(Stack) 나. 큐(Queue)
다. 데크(Deque) 라. 원형 큐(Circular Queue)

8. 스택 알고리즘에서 T가 스택 포인터이고, m이 스택의 길이 일때 서브루틴 AA가 처리해야 하는 것은?

```
T ← T + 1
If T > m Then
    Goto AA
Else
    X(T) ← Y
```

- 가. 오버플로 처리
나. 언더플로 처리
다. 입력 처리
라. 출력 처리

9. 스택(Stack)의 응용 분야와 거리가 먼 것은?

- 가. 인터럽트의 처리 나. 수식의 계산
다. 서브루틴의 복귀번지 저장
라. 운영체제의 작업 스케줄링

10. 스택 메모리에 대한 정보의 입출력 방식은?

- 가. FIFO 나. FILO 다. LILO 라. LIFO

11. 노드의 삽입 작업은 선형 리스트의 한 쪽 끝에서, 제거 작업은 다른 쪽 끝에서 수행되는 자료 구조는?

- 가. 스택 나. 큐 다. 트리 라. 그래프

12. 운영체제의 작업 스케줄링 등에 응용되는 것으로 가장 적합한 자료 구조는?

- 가. 스택(Stack) 나. 큐(Queue)
다. 연결리스트(Linked List) 라. 트리(Tree)

13. 자료구조 중 먼저 입력된 자료가 먼저 출력되는 형태로 헤드(Head)와 테일(Tail)을 입,출력 포인터로 사용하는 자료구조 형태는?

- 가. 스택(Stack) 나. 큐(Queue)
다. 데크(Deque) 라. 포인터(Pointer)

[정답] 7.가 8.가 9.라 10.라 11.나 12.나 13.나

14



[DB 8강]-자료 구조(선형, 비선형구조)

14. 다음 설명이 의미하는 것은?

- . 삽입과 삭제가 리스트의 양쪽 끝에서 발생할 수 있는 형태이다.
- . 입력이 한쪽에서만 발생하고 출력은 양쪽에서 일어날 수 있는 입력 제한과, 입력은 양쪽에서 일어나고 출력은 한 곳에서만 이루어지는 출력 제한이 있다.

가. 스택 나. 큐 다. 이중 스택 라. 데크

15. 선형 자료구조에 해당하지 않는 것은?

가. Binary tree 나. Dense list
다. Doubly linked list 라. Stack

16. 스택(stack)이 사용되는 경우가 아닌 것은?

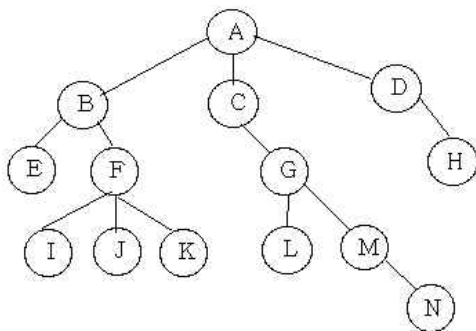
가. 인터럽트의 처리
나. 수식의 계산
다. 서브루틴의 복귀번지 저장
라. 스푼(pool) 처리

[정답] 14.라 15.가 16.라

15

[DB 8강]-자료 구조(선형, 비선형구조)

17. 다음과 같은 트리(Tree) 구조에서 기본 용어의 설명이 옳은 것은?

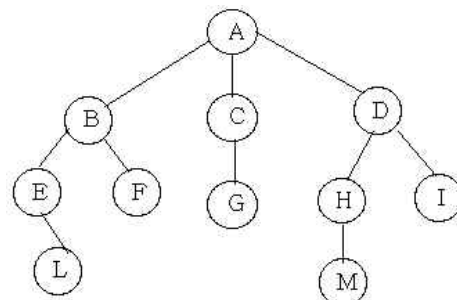


가. Node는 10이다.
나. Node의 차수(Degree of Node)는 4이다.
다. 레벨(Level)은 5이다.
라. 근(Root) Node는 N이다.

18. 깊이가 6인 이진 트리의 최대 노드수는?

가. 63 나. 64 다. 65 라. 66

19. 다음 Tree의 Degree와 터미널 노드의 수는?



가. Degree : 2 터미널 노드 : 4
나. Degree : 3 터미널 노드 : 5
다. Degree : 4 터미널 노드 : 2
라. Degree : 4 터미널 노드 : 10

20. 다음과 같은 중위식(Infix)을 후위식(Postfix)으로 올바르게 표현한 것은?

A / B * (C + D) + E

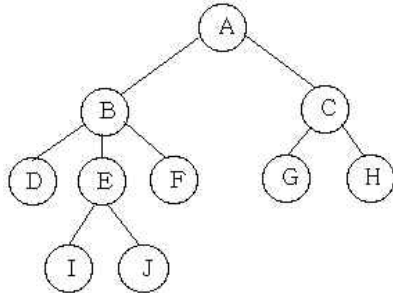
가. + * / AB + CDE
나. CD + AB / * E +
다. AB / (CD +) * / E +
라. AB / CD + * E +

[정답] 17.다 18.가 19.나 20.라

16

[DB 8강]-자료 구조(선형, 비선형구조)

21. 아래 그림에서 트리의 차수(Degree)를 구하면?



가. 2 나. 3 다. 4 라. 5

22. 이진 트리의 특성에 대한 설명으로 옳지 않은 것은?
(단, n_0 =단말 노드수, n_1 =차수 1인 노드수,
 n_2 =차수 2인 노드수, n =노드의 총수, e =간선의 총수)

가. $n = e + 1$
나. $e = n_1 + 2n_2$
다. $n = n_0 + n_1 + n_2$
라. $n_0 = n_2 + 2$

[정답] 21.나 22.라 23.나 24.가

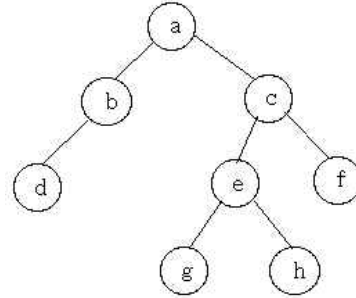
17

23. 다음과 같이 주어진 후위 표기 방식의 수식을 중위 표기 방식으로 나타낸 것은?

A B C - / D E F + * +

가. $A / (B - C) + F * E + D$
나. $A / (B - C) + D * (E + F)$
다. $A / (B - C) + D + E * F$
라. $A / (B - C) * D + E + F$

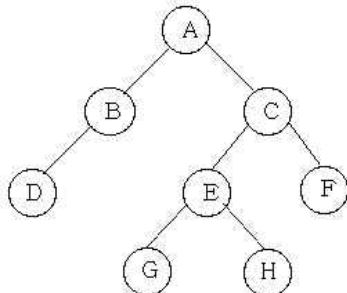
24. 아래 트리 구조에 대하여 Preorder 순서로 처리한 결과는?



가. $a \rightarrow b \rightarrow d \rightarrow c \rightarrow e \rightarrow g \rightarrow h \rightarrow f$
나. $d \rightarrow b \rightarrow g \rightarrow h \rightarrow e \rightarrow f \rightarrow c \rightarrow a$
다. $a \rightarrow b \rightarrow c \rightarrow d \rightarrow e \rightarrow f \rightarrow g \rightarrow h$
라. $a \rightarrow b \rightarrow d \rightarrow g \rightarrow e \rightarrow h \rightarrow c \rightarrow f$

[DB 8강]-자료 구조(선형, 비선형구조)

25. 아래 이진 트리를 후위 순서(Postorder)로 운행한 결과는?



가. ABCDEFGH 나. DBGHEFCA
다. ABDCEGHF 라. BDGHEFAC

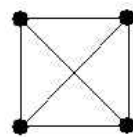
26. 이진 트리(Binary Tree)에서 발생하는 널(Null) 링크를 트리 운행에 필요한 다른 노드의 포인터로 사용하도록 고안된 트리는?

가. Knuth 이진 트리(Knuth Binary Tree)
나. 전 이진 트리(Complete Binary Tree)
다. B+ 트리(B+ 트리)
라. 스레드 이진 트리(Threaded Binary Tree)

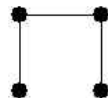
[정답] 25.나 26.라 27.다

18

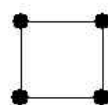
27. 다음 그래프 중 보기의 신장 트리(Spanning Tree)가 아닌 것은?



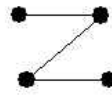
가.



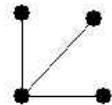
다.



나.



라.

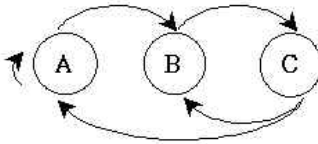


[DB 8강]-자료 구조(선형, 비선형구조)

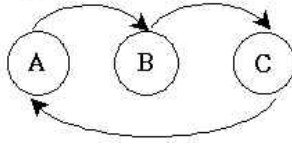
28. 다음 인접행렬(Adjacency Matrix)에 대응되는 그래프(Graph)를 그렸을 때 옳은 것은?

	A	B	C
A	0	1	0
B	0	0	1
C	1	0	0

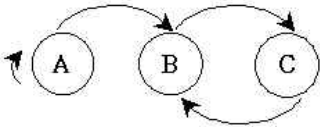
가.



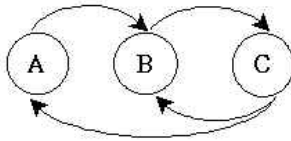
나.



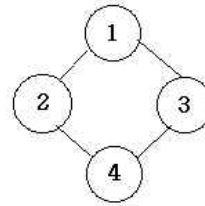
다.



라.



29. 보기와 같은 그래프에서 인접행렬이 옳게 된 것은?



가.

0	1	0	1
1	0	1	0
0	1	0	1
1	0	1	0

나.

1	0	0	1
0	1	1	0
0	1	1	0
1	0	0	1

다.

1	1	1	0
1	1	0	1
1	0	1	1
0	1	1	1

라.

0	1	1	0
1	0	0	1
1	0	0	1
0	1	1	0

[정답] 28.나 29.라

[DB 8강]-자료 구조(선형, 비선형구조)

[산-08년3월]

30. 큐(Queue)에 대한 설명으로 옳지 않은 것은?

- 가. 입력은 리스트의 한끝에서, 출력은 그 상대편 끝에서 일어난다.
- 나. 운영체제의 작업 스케줄링에 사용된다.
- 다. 오버플로우는 발생할 수 있어도 언더플로우는 발생되지 않는다.
- 라. 가장 먼저 삽입된 자료가 가장 먼저 삭제되는 FIFO 방식으로 처리된다.

[기-08년3월]

32. 순서가 A, B, C, D로 정해진 입력 자료를 스택에 입력하였다가 출력하는 경우, 출력 결과로서 가능하지 않은 것은?

- 가. D, A, B, C
- 나. B, D, C, A
- 다. C, B, D, A
- 라. B, A, D, C

[산-08년3월]

31. 다음의 산술식을 "Postfix" 표기로 옳게 나타낸 것은?

$$X=A+(B+C/D) \times E-F$$

- 가. $X=A+B+C/D \times E-F$
- 나. $XABCD/+E \times +F-=$
- 다. $=X-+A \times +B/CDEF$
- 라. $XABCDEF=++/\times-$

[정답] 30.다 31.나 32.가