

2008 학년도 2 학기 중간 시험 (1/2)				과 목 명	자료구조 (분반: 00□ )	학 과		학 년		점 수
담당교수	심 정 섭					학 번				
시험일시	10월	20일	월요일			성 명				

1. 다음은 추상자료형(Abstract Data Type, ADT)에 대한 설명이다. 맞는 설명을 **모두** 고르시오. (10점)

① ADT에서는 구현의 세부사항을 명시하지 않는다.

② ADT는 명확한 인터페이스를 사용함으로써 오류 가능성을 줄인다.

③ ADT를 사용하면 명시된 연산의 수행속도를 쉽게 알 수 있다.

④ ADT를 사용함으로써 알고리즘 또는 프로그램의 수행속도를 향상시킨다.

답:

2. 다음 의사코드(pseudocode)로 표현된 알고리즘 ABC의 시간복잡도를 최대한 정확(tight)한 O-표기법을 이용하여 나타내시오. (10점)

**Algorithm ABC (int n)**  
**for**  $i=1$  **to**  $n$   
 $i = i \times 2$   
 $XYZ(n)$   
**return**

**Algorithm XYZ (int n)**  
 $i = 1$   
**while**  $(i \leq n)$   
 $i = i \times 2$   
**return**

답:

3. 다음은 배열(array)과 연결리스트(linked list)에 대한 설명이다. 옳은 설명을 **모두** 고르시오. (10점)

① 배열은  $i$  번째 원소에 저장된 값을 최악의 경우  $O(1)$  시간에 변경할 수 있다.

② 이중연결리스트는  $j$  번째 원소와  $(j+1)$  번째 원소를 찾아 둘 사이에 새로운 원소를 넣으려면 최악의 경우  $O(1)$  시간이 필요하다.

③ 같은 수의 자료를 저장하고 있다면 배열이 연결리스트보다 항상 저장 공간이 덜 필요하다.

④ 환형배열(circular array)을 이용하면 삽입, 삭제 연산이 모두 최악의 경우에  $O(1)$  시간 필요하다.

답:

4. 1부터 100까지의 서로 다른 정수를 저장하는 벡터(vector) ADT를 크기 100의 환형배열(circular array)로 구현하였다. 그런데, 구현하려는 응용프로그램이 rank 0에 삽입하는 연산 및 rank 0에서 삭제하는 연산이 잦아서 이에 대해 효율적인 구현이 필요하다. 이때, 다음 각 물음에 답하시오. (25점)

(1) elemAtRank( $r$ )을 구현하는 방법을 100자 이내로 설명하시오. 단  $r$ 은 저장할 객체의 rank를 나타낸다.

(2) 초기에 비어있는 벡터에 다음 연산들이 차례로 수행되었을 때 결과(return 값이 있으면 return 값도 명시)를 (1)의 방법으로 그림과 함께 설명하시오.

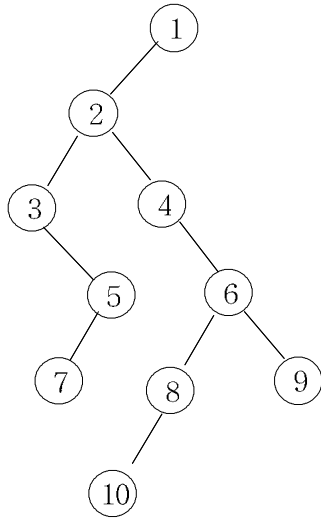
① insertAtRank(0,17)      ② insertAtRank(0,3)

③ insertAtRank(1,5)      ④ elemAtRank(2)

⑤ insertAtRank(1,99)      ⑥ removeAtRank(0)

⑦ insertAtRank(0,26)      ⑧ elemAtRank(3)

5. 강의시간에 설명한 방법으로 임의의 트리  $T$ 를 이진트리(binary tree)를 이용하여 저장하려고 한다.  $T$ 가 아래와 같은 이진트리에 의해 표현되었을 때 다음 물음에 답하시오. 단, 외부노드는 생략되었고, 노드 안의 각 수는 노드가 저장하고 있는 데이터이다. (35점)



(1)  $T$ 의 원래 모양을 그림으로 표시하시오. 단, 각 노드는 보기의 이진트리와 같이 노드가 저장하는 데이터(수)로 표시하시오.

(2)  $T$ 의 임의의 노드  $v$ 의 각 자식노드의 데이터를 차례로 출력하는 함수 `printChildren( $v$ )`을 구현하려 한다. pseudocode를 제시하고 시간복잡도를 설명하시오.

(3)  $T$ 의 각 노드  $v$ 에 대해,  $v$ 가 루트(root)이면서 노드가 가장 많은  $T$ 의 부분트리(subtree)의 높이를 출력하는 `height( $v$ )` 함수를 구현하는 방법을 제시하고 시간복잡도를 설명하시오.

(4)  $T$ 에 대해 후위순회(postorder traversal)를 수행하고자 한다. 저장된 이진트리에서 어떤 규칙으로 수행해야 하는지 방법을 제시하고 시간복잡도를 설명하시오.

6. 트리 ADT를 이진트리를 사용하지 않고 강의시간에 설명한 방법으로 구현하였다. 강의에서 설명된 트리 구현 방법을 이용하여 트리 내의 두 노드  $u$ 와  $v$ 의 위치(position)를 입력받아 두 노드가 형제노드(sibling)인지를 판단하는 불린(boolean) 함수 `isSibling( $u, v$ )`을 구현하려 한다. 트리의 구조를 그림으로 간략히 설명하고 함수의 구현 방법 및 시간복잡도를 설명하시오. (10점)