**자료구조 기말고사(2012.06.19)**

1. 다음 물음에 답하라
   1. Recursive 함수가 효과적으로 사용될 수 있는 3 경우를 제시하라.(10)

Shallow, efficiency(small overhead), clarity, pointer, trail recursion

* 1. Heap Sort가 Quick Sort에 비해서 유리한 점을 2개 이상 제시하라.(10)
     1. 항상 NlogN, 효율적인 메모리 사용(같은 공간 사용)
  2. Heap와 Binary Search tree의 다른 점 2가지를 제시하라.
     1. 상하 순서만 존재, 항상 complete binary tree 유지

1. Hashing의 a)장점을 설명하고, b) 충돌발생 가능성을 줄일 수 있는 방안 2가지를 설명하라.(15)
   * 1. Constant time 검색(5),
     2. 배열크기 증가, 자료특성에 맞는 hashing function 설계
2. Minimum Heap를 구성하기 위한 HeapUP(root, bottom)함수를 **iterative(비순환)**함수로 작성하라(15)

Void HeapUP(int root, int bottom) { int parent=(bottom-10/2;

While (root<=bottom) {

If (element[parent] < element[bottom]) break;

Swap(element[parent], element[bottom]);

Bottom=parent; parent = (bottom-1)/2; }

1. 순서 없이 작성된 일반 Binary Tree(**Btree)**에서 Postorder Traveral로 노드들을 방문하면서 입력된 키값(**inItem)**을 갖는 레코드를 찾아서 리턴하는 recursive 함수 RetrievePostOrder(TreeType Btree, ItemType inItem, Boolean found)를 작성하시오. TreeType과 ItemType classs는 강의자료에 정의되었던 것을 사용하고 추가적으로 필요한 class나 멤버변수나 함수가 있으면 정의하여 사용하라.(15점)

Found = FALSE; RetrievePostOrder();

Void RetrievePostOrder(TreeType Btree, ItemType inItem, Boolean& found) {

If (Btree !=NULL)

{

RetrievePostOrder(Btree->left, found);

RetrievePostOrder(Btree->right, found);

If (Btree->Info.key=inItem.key) {

inItem=Btree->info; found=TRUE;}

} }

1. 아래 테이블로 주어진 weighted graph를 기반으로 다음 물음에 답하라.(과정이 제시되지 않으면 점수가 부여되지 않음)
   1. Node A에서 Node D에 도달하는 path를 depth first search를 이용하여 구하라(10)

Stack={A}, [A] Stack={C,D}, [C] Stack={B,E,D}, [B] Stack={E,E,D}, [E] Stack={E,D},

[D] Stack={} 최종결과 A,C,B,E,D

* 1. Node A에서 Node E 사이의 최단거리 Path를 구하시오(15)

PQ={(A,A,0)}

A 🡪 A 0 VQ={C,D}

PQ={(A,D,10), (A,C,10)}

A 🡪 C 10 VQ={B,E}

PQ={(A,D,10), (C,B,20), (C,E,40)}

A 🡪 D 10 VQ={B,C}

PQ={ (C,B,20), (D,B,30), (C,E,40)}

C 🡪 B 20 VQ={C,E}

PQ={ (B,E,30), (D,B,30), (C,E,40)}

B 🡪E 30 VQ={D}

(D,B,30), (C,E,40)}

최단경로 A->C->B->E 거리 30

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Node | A | B | C | D | E |
| A |  |  | 10 | 10 |  |
| B |  |  | 20 |  | 10 |
| C | 10 | 10 |  |  | 30 |
| D |  | 20 | 10 |  |  |
| E |  |  |  | 10 |  |