9th assignment-2

컴퓨터전공

2013011491

안찬영

void adjust(int heap[], int root, int n)

{

int temp=heap[root];

int child=2\*root+1;

while(child<=n-1){

if((child<n-1)&&(heap[child]<heap[child+1])) child++;

if(temp>heap[child]) break;

else{

heap[(child-1)/2]=heap[child];

child = 2\*child +1;

}

}

heap[(child-1)/2] = temp;

}

좌우 자식 노드가 이미 히프 구조 일 때, 재조정하는 함수 입니다. root에는 재조정할 값의 배열 순서값이 설정되고, n에는 데이터의 개수가 저장됩니다.

임시 값에 root의 배열값을 저장한 뒤, child를 선언해, root의 왼쪽자식노드의 순서값을 설정합니다. 그 다음 반복문을 통해 오른쪽 자식노드가 왼쪽 자식노드보다 크면 child를 오른쪽 자식노드로 설정하고, root의 배열저장값이 child보다 크면 조정이 완료되고, 작을 경우 root의 배열값을 child의 배열값으로 바꿔준 뒤, child의 위치에 temp의 값으로 바꿔줍니다.

void heap\_sort(int list[], int n)

{

int i,j;

int temp;

for(i=(n-2)/2; i>=0; i--)

adjust(list,i,n);

for(i=n-1; i>=0; i--)

{

SWAP(list[0],list[i],temp);

adjust(list,0,i);

}

}

힙 정렬 함수입니다. 마지막 노드부터 adjust함수로 재조정 해서, heap구조를 만들고, 첫번째 배열값과 마지막 배열값을 바꿔준뒤, 마지막 배열값을 제외하고, 다시 재조정 해줍니다.

<결과>

C:\Users\Administrator\Desktop\결과2.jpg