



2023 Algorithm Study 3Week

[통계](#) [수정](#) [삭제](#)

sookyoung0620 · 약 5시간 전 · 비공개

0

2023 Algorithm Study

[▼ 목록 보기](#)

4/4



백준 2075

저번주에 문제의 주제를 인지하고 가는 게 올바른 방향으로 풀어나갈 수 있는 방향을 확실히 준다고 느꼈기 때문에 오늘은 힙, 힙정렬에 대해 알아보자

주요 키워드는 '완전 이진 트리' 인 거 같다! for 우선 순위 큐..

완전 이진 트리를 막 수월하게 설명할 수 있을 정도로 이해하고 있는 편은 아닌 거 같아 한번 찾아보았다

완전 이진 트리 : 마지막 레벨을 제외하고 모든 레벨이 완전히 채워져 있으며, 마지막 레벨의 모든 노드는 가능한



최대힙 : '부모의 값이 자식의 값보다 항상 큰' 조건을 만족하는 완전이진트리

최소힙 : '부모의 값이 자식 값보다 항상 작은' 조건을 만족하는 완전이진트리

내가 생각하는 힙 정렬의 포인트 :

1. 최대 힙 만들기 (부모노드와 비교하기)
 2. 루트 노드(아마 이때 가장 큰 수)와 배열의 마지막 값과 교환
가장 큰 값은 배열의 마지막에 저장
 3. 힙 다시 재 구조화
- > 반복!

이 문제의 경우 n번째로 큰 값을 출력해야하니까 n곱하기n-n 인덱스를 출력하면 되지 않을까 싶지만 잘 모르

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include <time.h>
#pragma warning (disable:4996)
```

```
void swap(int a[], int n1, int n2) {
    int temp = a[n1];
    a[n1] = a[n2];
    a[n2] = temp;
}

void Heap(int a[], int p, int size) {
    /*int parent = p;
    int left = p *2+1;
    int right = p * 2 + 2;

    if (left < size && a[parent]<a[left]) {
        swap(a, parent, left);
    }
    if (right < size && a[parent]<a[right]) {
        swap(a, parent, right);
    }
    if (left < size&& right < size && a[right]>a[left]) {
        swap(a, left, right);
    }*/
    int me = p;
    int parent = p/2;
    int bro;

    if ((me + 1) / 2 == parent) {
        bro = me + 1;
    }
    if ((me - 1) / 2 == parent) {
        bro = me - 1;
    }
    if (left < size && a[parent] < a[left]) {
        swap(a, parent, left);
    }
}
```

```

}
int main() {

    int a[1500],n;
    a[0] = 0;

    scanf("%d", &n);

    int size = n*n;

    printf("%d", size);

    for (int i = 0; i < size; i++) {

        scanf("%d", &a[i]);
    }
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            int p = i * n + j;
            Heap(a, p, size);
        }
    }
    for (int i = 0; i < size; i++) {
        if (i == n) {
            printf("%d", a[i]);
        }
    }

    return 0;
}

```

영망진창 나의 첫 코드(말이 첫 이지 코드를 아예 지우기 직전 코드^^)

문제점 1. 배열 선언.. 이중으로 할지 매우 고민함 Heap에서는 그냥 배열로 생각했으면서 main에서는 이중배열 선언
 문제점 2. 문제 이해 부족.. n곱하기n 형태로 입력받는 이유를 모르겠음.. 그래서 그냥 크기가 n곱하기n인 배열로
 문제점 3. 인덱스 착각... 배열은 0부터 시작하는데 순간 1부터 저장된다고 착각해서 Left Right 선언이 이상해?
 결론 : 이 코드를 보시는 분들 그냥 틀렸는데 뭐가 그렇게 헛갈렸지 생각하실 수도 있겠지만 이게 계속 섞여서 더
 저도 살려보려고 했지만 불가능하다고 판단하여 다 지워버림

```

> #include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include <time.h>
#pragma warning (disable:4996)

void swap(int* a, int* b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

```

```
void heapify(int arr[], int here, int size) {
    int left = here * 2 + 1;
    int right = here * 2 + 2;
    int max = here;
    if (left < size && arr[left]>arr[max])
        max = left;
    if (right < size && arr[right]>arr[max])
        max = right;

    if (max != here) {
        swap(&arr[here], &arr[max]);
        heapify(arr, max, size);
    }
}

void buildHeap(int arr[], int size) {
    int i, j;
    for (i = size / 2 - 1; i >= 0; i--) {
        heapify(arr, i, size);
        for (j = 0; j < size; j++)
            printf("%d ", arr[j]);
        printf("\n\n");
    }
}

void heapSort(int arr[], int size) {
    int treeSize;
    buildHeap(arr, size);
    for (treeSize = size - 1; treeSize >= 0; treeSize--) {
        swap(&arr[0], &arr[treeSize]);
        heapify(arr, 0, treeSize);
    }
}

void printArray(int arr[], int size) {
    int i;
    for (i = 0; i < size; i++)
        printf("%d ", arr[i]);
    printf("\n");
}

int main() {
    int arr[12250000], n;
    int size = 0;

    scanf("%d", &n);

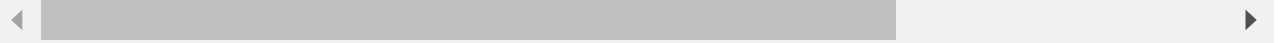
    for (int i = 0; i < n; i++) {
        size += n;
    }
    for (int i = 0; i < size; i++) {
        scanf("%d", &arr[i]);
    }
    heapSort(arr, size);
    printArray(arr, size);

    return 0;
}
```

여러가지 코드를 시도해봤으나 이중배열...?로 하는 방법을 전혀 모르겠다... 인터넷에 찾아봐도 안나오고... 그래



저번에 갑자기 뵈 받아서 목요일인가 이 과제를 조금 해놨었는데 안 했으면 큰일날 뻔했따.. 나의 패어와 유익한 스



운영진분들 기다려보세효 완벽한 코드 스터디 직전까지 도전하다가 완성되면 바로 c파일 올릴게용



윤수경



이전 포스트

2023 Algorithm Study 1Week-스터디 시간

0개의 댓글

댓글을 작성하세요

댓글 작성



Powered by
Stellate