[1]

오늘은 smp 6회차

동적할당 / 링크드 리스트 / 배열, 포인터 기출 배울 것이다.

[2]

목차

지난 시간 챌린지 문제 2개 풀이

동적할당 개념 설명 잠깐

링크드 리스트 개념 설명

기출문제 1개 풀이

[3]

<문제 설명>

이것을 해결하기 위해서는 우선 어떤 생각을 해야할까?

#1 학생의 성적을 입력받기 위한 배열, 그리고 반복문이 사용되겠구나

#2 가로에는 학생의 정보, 세로에는 점수에 따른 그래프 표현 = 이중 반복문이 사용되겠구나

#3 가로로 진행되는 반복문에서는 점수에 따라 공백 혹은 네모를 output으로 주면 되겠다.

<코드 output 보기>

어려운 이유 : 가로로 출력하면 쉬운데 세로로 출력해야됨

[4] [5]

<머지소트에 대한 설명> <https://www.youtube.com/watch?v=QAyl79dCO_k> 0분 ~3분 50초

머지소트는 크게 ‘merge’와 'sort’로 이루어져 있음. Sort에서는 배열을 절반으로 나눈 뒤 merge 함수를 부르는 작업을 하고

Merge는 나눠진 배열을 하나로 합치는 동작을 진행한다고 이해하면 됨.

[6] 동적 할당

동적 할당 설명하기

배열을 만들 때 원소의 개수를 항상 고정시켜야 했음 => 비효율적 => 어떻게 효율적?

‘힙’이라는 새로운 공간에 배열을 저장 => 따라서, 사용하고 난 뒤 free를 해주어서 할당된 공간을 다시 자유롭게 풀어주어야 한다.

#1 malloc

#2 calloc

[7] 기출문제

배열과 포인터에 대한 문제이다. “sorting”이라고 해서 이러한 문제가 자주 나오는데, 한 번 풀어보세요. 10분 드리겠습니다.

문제 설명

10분 휴식

[8] 링크드 리스트

링크드 리스트라는 것에 대해서 배울 것. 링크드 리스트에 대한 과제 1개, 그리고 다음주에는 어려운 과제를 하나 내드릴 것이다.

링크드 리스트에는 2가지 종류가 있다.

첫 번째의 단점은 무엇이겠어요? 이전 block을 가리킬 방법이 없다.

[9]

링크드 리스트 설명

링크드 리스트에는 크게 2가지가 들어간다. 노드의 "값” 그리고 다음 노드의 "주소”

배열에 비해서 좋은 점? = 만약 20개의 공간을 할당해놓고 10개의 공간만 쓴다면 배열은 10개의 공간이 낭비된 채 사용해야 한다. 하지만 링크드 리스트는 10개의 잉여 공간을 지워버리면 끝

[10]

그래서 아까, 링크드 리스트에는 두 가지 종류가 있다고 말.

[11]

이번 수업에서 직접 볼 것은 단방향 링크드 리스트. 제가 직접 코드를 구현해놨으니, 여러분들은 그것을 참조하여 양방향 링크드 리스트를 만들어보시면 된다.

배열은 서로 따닥따닥 붙어 있다면, 링크드 리스트는 멀리 떨어져있는 주솟값을 인위적으로 이어줄 수 있다.

[12]

배열과 링크드 리스트를 비교해놓은 표입니다.

“임의의 위치 접근” =

“임의의 위치 삽입/ 삭제” = 배열이 어려운 이유 = 가운데를 뻥 뚫는다면 새로운 배열을 선언해서 기존의 배열의 값들을 모조리 옮겨주는 작업이 필요, 하지만 링크드 리스트는 주솟값이 독립적으로 분리되어 있기 때문에, 가운데 부분을 뻥 뚫는다면, 양쪽의 링크드 리스트를 이어주면 그만.

“특정 값 데이터 찾기 / 데이터 정렬” = 링크드 리스트는 순차적으로 탐색해야함.

[13]

단방향 링크드 리스트 = 가장 간단한 링크드 리스트 종류이다.

장점 : // 왜냐, 포인터를 하나만 가져도 되니까

단점 : 포인터가 1개밖에 없어서, 다음 것에 대한 정보밖에 알 수 없음

[14]

링크드 리스트를 이루는 node

링크드 리스트를 관리하는 list가 존재

처음에 해야할 것은 list를 init(초기화) 하는 것이다.

List에 링크드 리스트의 첫번째 노드에 대한 정보가 있기 때문에, 링크드 리스트 전체를 지우고 싶다면 이렇게 쉬운 방법으로 지울 수 있다.

[15]

[16]

…

빈 리스트의 경우 코드가 바뀌어야 하냐? => 똑같아도 된다. 왜냐하면 우리가 처음에 list를 초기화할때 head 값을 NULL로 넣어줬기 때문에, temp의 next에 plist의 head값 (NULL)을 넣어주면 된다.

[17]

리스트의 “중간”에 삽입하는 경우, 중요한 것은 “이전 노드”에 대한 정보이다.

따라서, 먼저 pPre 노드에 대한 정보를 가진 채, 노드의 삽입을 시작한다.

…

“이후 노드”에 대한 정보가 아니라 “이전 노드”에 대한 정보를 알아야 하는 이유

= 단방향 링크드 리스트는 다음 노드에 대한 포인터만 갖고있음. 따라서 이후 노드에 대한 정보를 안다면, 우리가 넣을 node의 이전 node의 next를 우리가 넣은 node를 가리키도록 updata 해줄수가 없다.

[18]

단순히 코드로 나타낸 것

pPre에 대한 정보는, 함수에 들어오기 전 for문을 이용해서 “특정한 값”을 가진 노드의 포인터를 받아오면 되겠죠?

[19]

삭제하는 경우는 “삭제할 노드”에 대한 정보를 pos에 담아서 와야한다. (이것을 이용해서 함수 내부에서 for문을 이용해 pPre를 찾는다)

[20]

링크드 리스트를 코드로 나타낸 것

[21]

이번 주 숙제

[22]

다음 주에 숙제로 내줄 것 맛보기. 2018 11번째 lab 과제에서 가져왔다.

링크드 리스트, 구조체, 포인터, 문자열 개념을 모두 활용할 수 있어야지 풀 수 있는 문제.

따라서 다음주에는 진도에 맞추어 설명을 하고, 다음에 “양방향 링크드 리스트”랑 “문자열”에 대한 설명도 무조건 하도록 하겠습니다.

[23]

끝, 인데 만약 시간이 너무 많이 남았으면

+ 링크드 리스트 코드 같이 보기

+ 과목별 시험 팁

생명과학 : \*\*\* 교수님이 약간 수업하다가 다른 소리를 많이 하시는데, 거기서 문제가 좀 많이 나온다. 따라서 수업 시간에 필기 꼼꼼히 하고 다 외울 것 + ppt에 있는 내용은 무조건 다 외워야 함 + 영어단어 외우기

화학 : 분반마다 너무 달라서 뭐라 못하겠다. (페이스북 보면 어떤 분이 일반화학 수업을 계속 도와주시는데 그 수업 듣는거 추천)

미적분학 : 퀴즈 문제랑, 기출 문제 많이 풀어볼 것 +증명 문제

일반물리 : 물주 스스로 풀 실력이 되면 기출보면서 어느정도 공부할 수 있을 것.