1. 동적할당 다른 방법은 없나?
2. Free로 동적할당 해제를 해주지 않으면 어떻게 되나?
3. 연결리스트의 head 부분을 free로 반환하면 남은 연결리스트는 어떻게 접근하나?  
   애초에 해제하면 에러는 발생하지 않나?
4. 중간 부분을 해제해버리면 남는 부분은 어떻게 접근하나?
5. 애로우, 도트 뭐가 불편한지?
6. 구조체 함수 인수 전달 3가지 방법 장단점?
7. 노드 한 칸이 여러 개의 다음 노드를 가리키게?
8. 연결 리스트 삽입/삭제 애니메이션에 함수를 넣어서 이해하기쉽게 했으면
9. 김지
   1. 9페이 malloc, 하면 주소값을 반환, 변수에 그 값을 담아둔다거나 잃어버리면?
   2. 찾지못하면 돌려줄수가 없나?
   3. 컴파일러에 따라 다르지만 vc는 자동으로 free해주기 때문에 괘낞다고 생각
   4. 어차피 가만있어도 되는데 free 사용이유?
   5. 컴파일러에 따라 다르기 때문에, 남아 잇어 메모리에 충돌 유발
   6. 강솔: free 하는거 안하는거 표준?
   7. 하는게
   8. Vc는 안표준?
   9. 사용자의표준을 위해
10. 경환 15p, 다시 설명
    1. 이 부분은 알아서 적고
    2. 아까 free, vc는 쓸 필요 없다, 하나만 노드 생성해서 4번을 찾아가는데 tail을 null로 만들면 삭제?
    3. Follow만 만듬, 4번까지 감, 같이 가리키고 4번째의 tail을 널로, 5번을 가리키는게 땡기잖아
    4. 변수를 복사를 한다면 따로 끊기기 때문에 원래 있던 연결리스트는 끊기지 않는다.
    5. Tail을 널로 만들면 끊김?
    6. 끊겼다고는 하지만 삭제는 아님.
    7. 끊겼으면 5번 노드는 메모리상에?
    8. 참조를 할 수있지만 끊겨서 한 노드로 취급.
    9. 덕 추가
       1. 그게 아닌거같은데 질문자의 질문을 다시  
          head로부터 연결된 연결리스트, 중간의 노드가 next가 널이 되버리면 다음 노드들은? Follow, current 없음.
       2. 끊긴 시점부터 따로 관리가 됨.
       3. 그림으로 설명하자면.
       4. 찔럊;ㄴ 얀갈리스트? 게속 있음
       5. 자동으로 free?
       6. 모든 프로그램이 끝난후 free,
       7. 그럼 free 아닌데? 그건 vc에서 하는게 아닌데  
          크롬을 끌면 크롬이 관리하던 리소스는? 그게 해제되잖아, 램이 딸리면 프로세스 강종하잖아, 그거랑 같은? 그걸 vc가?
       8. vc에서 자동 free는 실행이 다 끝난후 free를 하지 않으면 메모리에 동적할당이 남아있는다. 다시 메모리를 불러올 때 충돌발생가능. Vc는 ~
       9. 실행이 끝난 후 어떤 컴파일러는 안해줌?
       10. 끝나면 리소스는 해제되지않나?
       11. For문 많이돌림 -> 리소스 부족 -> 강종
       12. 그건 정적, 동적은 free로 해제, 정적은 프로그램 종료 후 자동. 동적은 따로
    10. 김지
        1. 연결 리스트 할 수있는거 배열도 가능 장점?
        2. 배열보다 삽입/삭제 구조변경 간단. 배열은 인자값으로 바로 접근 가능, 연결은 하나하나 라서 느림
        3. 연결은 과정, ㅂ열은 삽입할 값과 인덱스만 있음ㄴ 된든데 뭐가 복잡?
        4. 배열 인덱스10, 하나 중간 삽입, 이거 알아서 하자.
        5. 연결 써서 단점?
        6. 말햇듯이 값을 찾을 때 주소로 바로 가지 못해서 처음부터 가야함, 단일이라 그럼. 이중,원형이면 특히 원형, 값이 맨 뒤에 있으면 간단하게 찾을수있음.
    11. 솔
        1. 전 질문, 메모리 해제 안하고 프로그램 종료하면 vc 말고 다른 컴파일러는 해제 안해서 메모리에 충돌이 일어난다는데 충돌?
        2. 동적할당했을때 아무값도 입력하지 않았는데 값이 있다, 나머진 몰라
        3. 임의로 다른 메모리 주소를 들어가서 접근해봤니?
        4. ㄴ
        5. 다른 메모리에 임의로 접근하면 초기화되있다 생각하는데 맞나
        6. 쓰레기값
        7. 쓰레기값이면 충돌해도 덮어씌움되잖아
        8. 동적할당하여 연결 생성, 생성하고 free하지 않음 다시 켜서 할당하면 넣지 않았음에도 넣어져있다는 가능성
        9. 그값 쓰레기잖아. 쓸거? 내용도 안읽고 쓸라고?
        10. 몰라
    12. 솔
        1. 프로그램 돌림, 계속 동적할당, 끝까지 free안하고 계속 하면?
        2. 메모리 ㅓㅁ어가서 스택오버플로우 발생
        3. 프로그램 꺼짐? 발생해도 계속 실행?
        4. 발생하면 실행오류 -> 종료
        5. 끝까지가 메모리를 다 쓰면 소.오?
        6. 8기가 메모리, 4기가 할당 가능. 모든 프로그램이 다 써버리면 다 죽나?
        7. 아마 죽나
        8. 경험해봄? 다 모든 죽는거 블스만 뜸.
        9. 다시 보거라
    13. 김지
        1. 궁금한건데 9, 동적할당하는데 초기화할 때 문제 발생, 뒤에 확인하는 코드 필요>
        2. 실패하면 amlloc에서 0 반환. 그걸로 조건식
    14. 강솔
        1. 실패하는 경우?
        2. 메모리 다 찼을 경우
        3. 실패하고 다시 시도하면 스,오?
        4. 다시 생각해보니 스,오 발생하지 않고 할당이 안됨. 일어나지 않음
        5. 아무리 할려도해도 할당안됨? 모든 프로그램이 할당할수업슨ㄴ 상태? 하나가 다 할당해버리면? 1명이 악의적으로 할당해버리면 컴퓨터 못씀?
        6. Free로 해제할수 없다면 사용불가
        7. 다른ㅇ 프로그램이 free를 해서 써야하나 그래야 안죽고 쓸 수 있나?
        8. ㅇ
        9. 하 알겠다
    15. 고기
        1. Free, 할당하고 free하고 한번더 free하면?
        2. ~~좆되지~~. 오류 발생 컴파일러에서 안된다고 해서 실행이 안됨.
    16. 김지
        1. 발표자가 동적 도엊ㄱ 거리는데 정적과의 차이
        2. 정적: 한번 할당하면 변경 불가능, 끝날때 자동으로 메모리를 반환,  
           동적:자유자재 할당가능, freee 를 통해서 반환해줘야함  
           장단점, 동적은 메모리상 자유, apahj리 쌓이면 할당안됨  
           정적은 고정되있어서 위험없음
        3. 동적도 해제하지않으면 vc에서 해준다는데 정적도 원하는만큼 여러 개 선언하면됨. 차이? 뭐가 동적?
        4. 글치만 메모리의 변경이 불가능, 지울수없음. 동적은 free 로 지울수있음.
        5. 동적은 메모리를 늘린다거나 수정도 가능?
        6. ㅇ, realloc.
    17. 경환
        1. 변수 선언되면 메모리의 크기만큼 선언받는다. 그 변수들은 정적? ㄷㅇㅈ?
        2. 정적
        3. int형이면 메모리에 어느 크기만큼? 최대 한도? 어느만큼 선언가ㅡㅇ?
        4. 메모리의 최대한도.
        5. 엄청나게 많이 선언됬닫 치고 얼마마늨므의 메모리 감당가능?
        6. 컴터에 따라.램이나 여기저기
        7. 4기가 램, 컴퓨터 키고 여러 개 돌리는데 2기가, 정적 메모리에 2기가만큼 담을수있나?
        8. 남으면 담을 수 있지 않겠니
        9. 남았다치고 꽉 차면 동적할당 가능?
        10. 동적할당은 가상 메모리, 따로 할당이 되지 않나 싶기도 하고 모른다.
    18. 강솔 추가
        1. 가상메모리?
        2. 가상의 메모리.
        3. 정확하게
        4. 조사하던거라 머리가 이해를 못함. 카페에
    19. 덕현
        1. 10, 연결 설명할 때 단일만 했는데 원형,이중 있다 함. 단일, 이중, 원형 차이, 장단점?
        2. 단일 이중 비교, 단일 next만 , 이중은 prev도 있어서 앞의 노드에도 접근 가능.
        3. 이중, 원형 비교, 우너형은 이중에서 마지막 노드, 처섭ㄴ째 노드가 이어져있음. 맨 끝에 있음 갈 필요없이 그냥 접근하면 된다.
        4. 간단하게 다시
        5. 단일 한쪽만, 이중 반대쪽으로, 원형 이중 + 처음 긑 이어져있음.
        6. 어떤 장점?
        7. 처음에서 뒤로가서 접근가능. 짧게
        8. 어떤 데이터가 어딨는지 아는방법?
        9. 자세ㅔ히 살펴보지 않아서 모름.
    20. 고기
        1. 2, 구조체 의미상 서로 연관? 뭔 의미?
        2. 사원의 이름, 월급, 번호 같으면 연관있잖아 그게 의미상 연관이지. 근데 자료형이 다를 수 있으니 구조체르 통해서 하나의 이름으로 묶어서 사용가능.
    21. 경환
        1. 사원있다 치고 회사 정보 구조체 만듬. 이름, 년도, 직원수있다치고, 사원이라는 구조체안에 회사라는 구조첼르 넣으수있니?
        2. 다르면 대입 불가능.
        3. struct안에다 구조체 못들어감?
        4. 구조체안에 구조체 말이야 임ㅁ
        5. 해보진않음, 가능할거야 구좇 ㅔ자료형이기 때문에 자료형이 들어가니까 되지 않을까?
        6. 강솔
           1. 함수 가능?
           2. ㄴ
           3. 자료형이 들어가서ㅏ 함수 ㄴ
    22. 고기
        1. 도트, 애로우, 도트 사용하면 애로우보다 복잡 어떻게 복잡?
        2. ~~몰라 ㅅㅂ랑~~ ㅇ 도트 연산
           1. 덕현 애로우, 도트 차이아냐?
           2. ~~도트~~ 구좇
           3. 하나에는 붙어있고 하나에는 붙어있지않음, 차이?
           4. 값을 대입할 때 원래 변수에서 주소 연산자를 사용하는 다른 자료형들과 문자열들과
           5. Employee.name? 무슨값?
           6. 문자열.
           7. 문자열 마지막에 뭐가 들어감?
           8. 문자의 배열, 문자열의 차이 아니?
           9. Null 있고 없고
           10. 네임이 어떤 값을 가지냐?
           11. 어떤 타입을 가지냐고?
           12. 주소의 값.
           13. 그래 주소값, 이상이다.
    23. 정환
        1. 연결리스트, 원형, 이중 있다 함. 다른 연결리스트 찾아본적?
        2. 없다. 종류는 단일, 원형, 이중이 있다해서 3개만 말함.
        3. 청크드 연결?
        4. 차장봐
    24. 덕현
        1. 구조체 안에 구조체 들어갈수있냐는 질문, 트리 들어봄?
        2. ㅇ
        3. 트리가 c언어에서 어떤 식으로 구현?
        4. ㄴ
    25. 고기
        1. 구조체도 자료ㅗ형, 구조첼르 2개선언해서 그게 대입이 됨?
        2. 선언했을때 잘형이 같으면 대입이 되지만 다르면 안됨.
        3. 캐스팅 연산자 안도미?
        4. ㅇ, 멤버는 따로 대입이 됨.
    26. 강솔
        1. 멤버 같음, 이름만 다름. 호환됨?
        2. 해보지안흠. 아마 되지않을까
        3. 아까 절대 ㅗ바꿀수없다고
        4. 뭘?
        5. 아닌가 그건 이거일 때 ㅇㅇ
        6. 하나는 샐러리 위로 올리고, 그러면 됨?
        7. ㄴ
        8. 왜?
        9. 다른 구조체이기 때문에 되지 않음.
        10. 아깐 된다고
        11. 앞서 말했듯이 똑 같은 궂체일 경우에 1은 a,b 만들고 2로 c,d 만들면 a,c는 대입이 안됨.
        12. 1, 2 ㄱ각각의 변수끼리 호환이 안됨? 캐스팅도 안되, 값도 안되?>
        13. 구조체는 안되지만 멤버로 각각 넣으면 됨.
        14. 글면 구조체 선언해서 메모리에 올리면 어떻게 메모리에?
        15. 네임 20바이트, 인트형 샐러리 4바이트
        16. 똑같잖아. 이름만 다르잖아 호환안됨?
        17. 일단 타입캐스팅해서 호환된말이
        18. ㄴ
        19. 왜
        20. 구조체의 타입이 다르기에 구조체란 변수를 따로 만드는것이기 때문에
    27. 경환
        1. 메모리 그림그려줘 5페이지 주소값도 메모리 연속으로 이어짐?
        2. Int가 위로 가있으면?
        3. 반대로 ㅇㅇ 해서 천하고 천사하고 천이십사하고
        4. 자료형이 어디서부터 어디까진 인트, 캐릭터 이런거 있ㄴ c에서 1에서 100은 인트, 이런거
        5. 햐주진 않지만 구조체 선언하고 각각 멤버에 대한   
           바이트 크기에 따라서실행잉 되고
        6. 강솔
           1. 구조체 사이즈?
           2. 24임
           3. ㄹㅇ?
           4. 글면 아까 했던 얘기 연장선 쌓이잖아, 이름만 다르잖아. 캐스팅 안되냐고 ㅅ빌아 진짜 아오 내가 진짜 뚝배기 깨버릴까 ㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋㅋ 황금나선!아니아니아니아니아나인 이름만 ㅏㅂ꿔ㅏ
           5. 1,이 저려면 2는?
           6. 똑같잖아, 캐스팅팅하면 안되냐고
           7. 그러면 int, float 왔다갔다 해보면? 비슷한 개념이잖아
           8. 책은 다른 구조체는 대입이 됮 않는다 라고
           9. 해봄?
           10. 똑같이 만들어서 해보진 않음
           11. 다르게 하니 안됨?
           12. 안된다고 생각.
           13. 캐릭터ㅏ 18, 인트 4 크기?
           14. 24. 구조체 저장방식. 큰 자료형에 따라 저장. 큰 단위에 저장. 캐릭터 = 1, 인트 =4 , 18이면 2칸남아있고 인트 추가되서
           15. 그럼 2바이트 낭비? 따로 변수 안하고
           16. 주소를 따라서 이 값 다음의 주소가 이 주소이기 때문에 이 주소의 값을 넣고 값을 참조를 하면 뜨지 않을까
11. 코멘트
    1. ~~정환~~
       1. 피피티 완성되가는데 뭔가 조금 이상하긴하지만 좀 더 노려가면 될 듯. 발표도 코멘트 받았던거 고쳐가는데 조금 부족한거 같고 준비 열심히 했음 다음도
    2. 고기
       1. 예ㅓㄴ보다 나아짐. 이전보다 나이짐 피비티 보면 코드 구분힘듬. 6보면 한줄짜리 막 되있고 구분이 안되서 보기 힘듬. 질문하면 단답식말고 이유도 같이 설명했으면
    3. 강솔
       1. 일단 확인되지 않은 사실에 확실하고 해보고 안해봤음 너무 그 뭐야 강력하게 주장안했으면, 어 아까 말햇듯이 여전히 한번더 질문하게 만들고 단답형으로 대답해서 한번더 질문하게 만들고 5, 예제 좋지 않다. 도트, 애로우 차이 도트는 불편하다 했는데 그거 하라고 만든 예제아니였으면 ep 썻으면 이해 쉬웟을 듯. 연결 애니 종이엔 생략되있어서 피피티봐도 그냥 얘기하는데 어디말하는지도 모르겠고 아이템도 없고 설명하는데 받아들이는데 이해힘듬. 포인터 쓰지 좀. 어 이상
    4. 덕현
       1. 처음 시작 딱딱, 부자연, 선배 마랫듯이, 구분이 안되있고, 섞여서 들어가있어, 뭐가 함수고 뭐가 구조첸지 몰라, 들으면서 어 저게 맞나 싶은 생각이 자주 듬, 예제 자주보면 연결보면 싹 다 노드인데 설명 듣다가 어디하는지 모름. 빠른 감도 있고 정확히 어디하는지 없고 설명하다 페이지 넘어가고 많아서 어느순간부터 다른 페이지를 보고있었음. 고쳐라
    5. 진우
       1. 저번 봣을 때 몇번 못봤는데 굉장히 나이진 것 같고 나름 괜찮은것같고 프로그래밍할때 규약같은게 많은데 첫글자 대문자 뭐시기 낙타 그거 알아서 적고. 규약이 ㅇㅆ는데 8, 함수 이름이 이상해서 놀람, 고치고, 상수 잘했고 함수부분은 다 붙어있고 떨어져있고 하나로 통일해서 사용하는게 맞는듯.
    6. 경환
       1. 가독성 앞에서 다 얘기함 하말 없다. 물어보면 그래도 모르겠다 안하고 추측답변이라도 해서 좋다 생각함. 결과는 안좋지만 어떻게든 말할ㄹ고 하는게 좋다. 가독성만 어/떻게.
    7. 김지
       1. 말은 잘하네 질문도 에아니요만 아님 잘하는거 같고 발표만 책 읽는거 같다. 어떠한 내용에 발표하겠다 목차 없고 구조체하면 말해주겠다 하고 정리를 하거나 끊으면 좋은데 혼자 책 읽는것같다. 듣던말던 내용만 말하는, 그런 느낌. 발표 도중에 따라가기 힘듬, 이해도힘듬. 공부 했고 잘한거같다. 이상ㅇ