[1]

자, 이번 주차는 프로그래밍과 문제해결 smp ‘4주차’입니다. 오늘은 반복문이랑 재귀함수, 그리고 파일 입출력 개념에 대해서 다룰 것이구요. 반복문을 위주로 문제를 풀어볼거고 재귀함수는 간단하게 설명하고 같이 문제를 풀어보려고 합니다. (하노이탑 문제 보셨죠?? 봤다면, 제가 그 부분이 많이 중요하다고 생각을 해서, 정확한 코드 흐름 같은거는 다다음주 수업 때 명확히 짚어드리도록 하고, 오늘은 재귀에 대한 기초적인 맥락은 아신다고 제가 가정하고 하노이탑보다는 쉽지만, 재귀 함수를 이용해볼 수 있는 문제를 몇 개 가져왔어요) 그냥 다다음주에 본다고 이야기) 파일 입출력은 PPT에 관련한 내용이 잘 나와있으니, 예제를 한 번 읽어보시고, 혹시 잘 모르겠다 싶으면 톡방에 올려주세요. 그러면 다다음주에 수업 준비해올 때 같이 설명해드리도록 하겠습니다.

[2] 자, 그래서 우선 반복문 강의 PPT중 짚고 넘어가야될 사항을 짧게 10분 정도 알아볼 것이에요.

그러고 나서, 제가 기초적인 반복문 문제를 여러 개 만들어왔거든요? 시험보다 훨씬 쉬운 수준으로 만들어왔습니다. 이중에 2개정도, 혹은 아직 잘 모르시는 것 같다 싶으면 3개 모두 코딩으로 5분씩 주고, 풀어보도록 할게요.

그러고 시간이 남는다면 반복문과 관련한 기출문제, 같이 풀어보도록 할 것이에요.

그리고 10분 정도 쉬었다가,

이번 어싸인 제가 어제 잠깐 읽어보고 시작 메뉴랑, 확률 구현 관련한 내용을 코딩해왔거든요? 그래서 그거 보면서 어싸인에 대해 간략히 설명 드릴거에요. 여기서 분기문에 관련한 설명도 같이 드릴 수 있을 것 같아서 제가 만들어왔어요.

다음 어싸인 부터는 제가 시간이 된다면 가급적 아예 구현을 해와서 같이 review 하는 시간을 가질 수 있도록 노력해볼게요. 대신 완성된 어싸인 코드는 제가 드리지 않을거에요.

마지막으로 재귀함수, 파일 입출력 설명

[3]

반복문에는 for / while / do while이 있다. 그런데 do while은 사실 쓸 일 아예 없음. 그냥 while문으로 구현하면 되고, 혹은 while문이 실행되기 전에 할 내용을 먼저 실행시키고 while문 쓰면 됨.

for문과 while문도 어느정도 비슷한데, for문에 저렇게 쓰면, while(1)과 똑 같은 작용을 함. 근데 코딩을 하다보면 어느 때 for문, 어느 때 while 문을 쓰면 조금 더 편하게 구현되겠다 하는 느낌은 있는 것 같아요.

나머지 내용은 그냥 기초적인 사용 예제와 관련된 내용인데, 질문이 없었어서.. 제가 스킵했습니다.

[4]

반복문을 배울 때, 가장 기초적으로 해보는게 피라미드 만드는 예시라고 생각이 되는데요, 저도 smp 시간에 이 문제 풀어봤었던 것 같아요. 한 번 왼쪽 문제 먼저 풀어보시겠어요?

왼쪽 문제를 모두 잘 푸신다 싶으면, 2번 문제는 그냥 생략하도록 하겠습니다. 5분 드릴게요.

[5]

다음으로는, 자연수 N을 소인수분해 하는 프로그램을 만들어보세요. 저는 while문으로 구현했는데, 반복문 중 아무거나 사용해서 구현하셔도 좋습니다. 5분 드릴게요. (예시 150 입력하기)

[6]

마지막으로, 간단한 기출문제를 한 번 풀어보고 잠깐 쉬는 시간을 가지도록 할게요.

이거는 앞 장에 나오는.. 한 2점 정도 짜리 쉬운 문제인데, 한 번 눈으로 풀어보세요. 5분 드릴게요.

2.4 x = 46

Do-while 같은 경우는 검증이 뒤에 있다는 식으로 말을 하죠? 그래서 x = 46을 만든 뒤에 x와 y의 대소비교가 이루어집니다. 따라서 x = 46이 되어요.

2.5 x = 10 , y = 25

보면, 제가 예전 시간에 x가 짝수인지, 홀수인지 판별하는 식은 되게 자주 만나게 될 것이라고 말씀드렸었죠? If 문의 조건을 보면, x가 짝수일 때 continue 즉 for문의 초기로 돌아가라고 말을 하고 있습니다.

따라서 y에는 1 + 3 + 5 + 7 + 9 홀수만을 더한 값이 들어갈 것이고, 이 값은 25가 됩니다.

(쉬는시간)

[7]

자, 이제 어싸인1에 대해서 간략한 개요 설명.

어싸인 pdf 문서를 잘 읽는 것이 굉장히 중요하다. (특히, 처음 프로그램을 작성해보는 입장에서, 어느 함수가 어떻게 이용되어야 하는지 감을 잡기 어려울텐데, pdf에 사용해야하는 함수 목록을 참고해 만드는 것도 큰 도움이 될 수 있다.)

이제, 개요 설명.

Switch문 이용하라고 되어있으니,

…

1.번을 이용해 게임이 시작이 되면, start\_game 함수를 이용해 게임을 시작하시면 될 것입니다.

저는 1번 목록에 게임 시작은 구현을 하지 않았구요. 대신 어싸인 파일을 읽어보니 1루타, 2루타, 3루타.. 이렇게 확률 계산하는게 있어서 해당 부분을 구현해놓았다.

100% 확률 중, 각각 case에 대한 확률이 정해져 있으므로, 지난 시간에 설명한 ~

[8]

자, 그리고 한 친구가 저한테 질문을 준 사항이 있었는데요, 제가 그 때에는 그냥 제가 생각하는 방식대로 해결해주고 설명을 해드렸었는데, 한 번 공식적인 자료를 찾아와봤어요.

(main에 srand 두고 실행) 다른 결과

(함수에 srand 두고 실행) 같은 결과

이게 왜 그러냐면요, srand 함수의 동작 원리부터 알아야 합니다.

라이브러리 내부 사전에는 난수 생성을 위한 엄청 큰 표를 가지고 있는데요, 그걸 이용해서 우리가 난수를 뽑아냅니다. 이 하나하나의 표를 ‘시드’라고 하고, srand(time NULL)을 이용해 우리가 랜덤한 시드를 뽑아내는 작업을 하고 있는거에요.

시드가 결정되면, 그 다음부터 해당 시드 삿전에 저장되어 있는 값이 순서대로 호출이 되는 거에요. 시드 표 자체는 정해져 있더라도, 어느 시드를 뽑느냐는 우리가 시간을 이용한 확률변수로 나타나는거니까 결국 random이 되겠죠?

그런데, time의 해상도는 1초이다 라는 말을 합니다. 이게 뭐냐면, 결국 우리가 함수에 srand()를 넣고 계속 초기화 시켜주었는데, 같은 값이 나오는 이유는, 이 main함수에 있는 for문이 ‘1초’안에 모두 실행되기 때문에, 계속 같은 시드의 첫번 째 수를 뽑아주고 있는거에요.

그래서 제가 여러분들한테, 이 사실을 보여드리기 위해서 이제는 scanf를 놓아서, 시간 딜레이를 줘볼게요.

(함수 안에 scanf 줘보기)

보세요. 이제 다른 값이 출력되죠?

제가 급하게 알아와서 설명 드리기는 했는데, 그냥 한 귀로 듣고 흘리셔도 되는 내용이니까, 한 번쯤 들어보면 좋을 것 같아서 제대로 설명 드렸습니다.

[9]

피보나치 수 설명 => 피보나치 수 구하는 과정 설명 => 이런 것을 재귀함수라고 한다.

재귀함수에는 크게 두 가지를 설정하면 된다.

1 종결조건 / 피보나치의 경우 0과 1

2 연결식 / 점화식이라고도 하는데, Fn = Fn-1 + Fn-2로 구해진다.

코드 설명

자 그러면, 재귀의 본질적인 원리는 하노이 탑 예제를 보면서, 다음 시간에 복습하며 알아보는 것으로 하고, 우리는 한 번 재귀 함수를 이용한 코드를 짜볼 거에요.

[10]

재귀 함수를 이용해 피라미드를 출력해보세요. 여러 분들 아마도, 재귀 함수를 공부하신 적이 없다면 익숙하지 않아서 이거 어려울 거라고 생각을 하거든요? 한 번 어떻게 하면 피라미드를 출력할 수 있을 지 생각해보세요.

종결 조건이 무엇인지 / 그리고 재귀 함수 간의 연결은 어떻게 되는지, 보면 숫자가 하나씩 늘어나고 있죠? 재귀 함수를 부를 때, 숫자를 하나씩 증가시켜주면 될 거에요.

역재귀 함수를 이용해 피라미드를 출력해보세요.

[11]

Fscanf 반환값 = 읽어들인 데이터의 수 // 아마도 이 경우 char형에 해당하는 1 바이트를 반환할거에요.

EOF = end of file / file의 끝에 다다르면 fscanf는 EOF를 반환한다 (파일이 끝났음을 알려줌)

“fclose” 꼭 해줘야한다. (시험에 꼭 나옴!!)

[12]

Fopen을 “w” 타입으로 할 경우, 존재하는 파일을 열면 존재하는 파일의 내용을 초기화하거나 / 존재하지 않는 파일을 열면 컴퓨터가 알아서 그 파일을 해당 위치에 만들어줍니다.

그래서 사실 if (outfile == NULL) 부분은 실행되는 경우가 거의 없다고 보시면 돼요.

[13]

방금, 제가 “W” 타입으로 존재하는 파일을 열면 존재하는 파일의 내용을 초기화한다고 하였죠? 근데 만약, 존재하는 파일을 여는데, 그 파일에 내용을 이어붙이고 싶다면 “a” 타입으로 열면 됩니다. “a”는 append의 줄임말로, 이어 붙인다는 뜻을 가지고 있어요.

사실, 파일 입출력과 같은 부분은 별로 중요하지 않은 내용이에요. 예제를 읽어보고 사용 방법을 익히기만 하면 필요할 때 내용을 찾아보면서 코드를 작성하면 되거든요.

그런데, 그렇기에 시험에 나오면 헷갈릴 확률이 커요. “꼭” 시험 전에 파일 입출력 부분 확실하게 외우고 들어가시는거 추천드릴게요. 제가 중간고사때 이 부분에서 뭐 하나 실수해서 감점 당했던 것 같아요.

관련해서는 여러분들 PPT 22페이지 ~ 25페이지에 보면 예제 코드가 있거든요?

이 부분만 정확히 이해하고 가시면 시험 때 파일 입출력과 관련한 문제를 틀리실 일은 없을거에요.

저도 시험 공부할 때 이 예제코드를 거의 외우다시피 공부하고 갔거든요. 시험때 갑자기 흐름이 끊겨서 헷갈리긴 했지만 ㅠㅠ

[14]

마지막으로, 제가, 파일 입출력과 반복문과 관련해 적절한 숙제를 생각하다가, 이전에 저희 lab 시간에 과제로 나왔던 문제를 찾아오게 되었습니다.

[15]

끝,

제가 이번 smp를 조금 급하게 준비하느라 부족한 내용이 있었던 것 같은데

‘재귀함수’ 와 관련해서 꼭 알아야 하는게 하노이 탑 알고리즘 이거든요. 오늘은 단순히 재귀함수를 이용한 프로그래밍을 표면적으로 배웠다면, 다다음주 시간에는 이 내용을 설명해드리면서 재귀함수의 동작 과정에 대해서 알려드리도록 할게요.

파일 입출력과 같은 경우는, P