자료구조 과제 1

연습문제 (1)

**1.8** 다음의 함수를 각각 O(Big-Oh)-표기법으로 표현하시오.

a) $10N^2-3N+9 = O(N^2)$ $(c=11, N_0=2)$

b) $2N^2+N\log N+5N = O(N^2)$ $(c=3, N_0=10)$

c) $8N^3+3N+5 = O(N^3)$ $(c=9, N_0=3)$

d) $2^N+N^3+5 = O(2^N)$ $(c=2, N_0=10)$

**1.10** 다음 루프의 수행시간을 $\Theta$-표기법으로 표현하시오.

```
01 int s=0;                          → 1
02 for(int i=0; i<N; i++)            → N+1
03     for(int j=0; j<N; j++)        → N(N+1)
04         s+=N;                     → N²
                                     ----------
                                     2N²+2N+2
```

$2N^2+2N+2 = O(N^2)$ $(c=3, N_0=3)$

$= \Omega(N^2)$ $(c=2, N_0=1)$

$= \Theta(N^2)$

**1.11** 다음 루프의 수행시간을 $\Theta$-표기법으로 표현하시오.

```
01 int s=0;                          → 1
02 for(int i=0; i<N; i++)            → N+1
03     for(int j=0; j<i; j++)        → N(N+1)/2   1+2+...+N
04         s+=j;                     → (N-1)N/2   0+1+...+N-1
                                     ----------
                                     N²+N+2
```

$N^2+N+2 = O(N^2)$ $(c=2, N_0=2)$

$= \Omega(N^2)$ $(c=1, N_0=1)$

$= \Theta(N^2)$

연습문제(2)

1.19 다음의 메소드에 대해 f(4)를 호출한 결과는?

```
01 public static void f(int n) {
02         System.out.print(n);
03         if (n>0) f(n-1);
04 }
```

f(4)호출 → 4출력 → (4>0) 이므로 f(3)호출

→ 3출력 → (3>0) 이므로 f(2)호출.

→ 2출력 → (2>0) 이므로 f(1)호출

→ 1출력 → (1>0) 이므로 f(0)호출.

→ 0출력 → (0=0) 이므로 f(0) 종료

→ f(1) ~ f(4) 까지 동일하게 종료.

<console>

4210

1.20 다음의 메소드에 대해 g(4)를 호출한 결과는?

```
01 public static void g(int N){
02     if(N>0) g(N-1);
03         System.out.print(N);
04 }
```

g(4) 호출 → (4>0) 이므로 g(3) 호출

→ (3>0) 이므로 g(2) 호출.

→ (2>0) 이므로 g(1) 호출.

→ (1>0) 이므로 g(0) 호출.

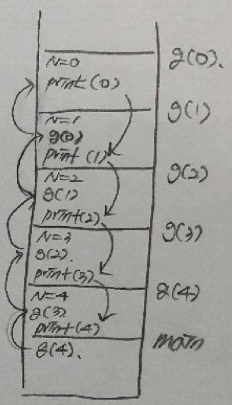
→ (0=0) 이므로. 0 프린트. 후 g(0) 종료.

→ 1프린트 후 g(1) 종료

→ 2프린트 후 g(2) 종료

→ 3프린트 후 g(3) 종료

→ 4프린트 후 g(4) 종료




<console>

01234

1.21 다음의 메소드에 대해 h(4)를 호출한 결과는 ?.

```
01  public static void h(int N)
02      System.out.print(N);
03      if(N>0) h(N-2);
04      System.out.print(N);
05  }
```

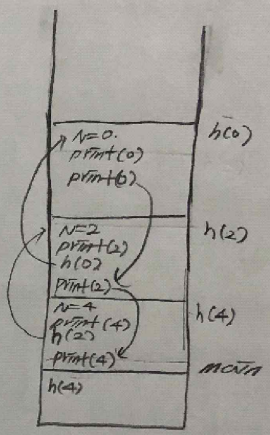
h(4) 호출 → 4프린트 → h(2) 호출.

→ 2프린트 → h(0) 호출 → 0프린트 → 0프린트 후 h(0) 종료.

→ 2프린트 후 h(2) 종료 → 4프린트 후 h(4) 종료.

<console>.

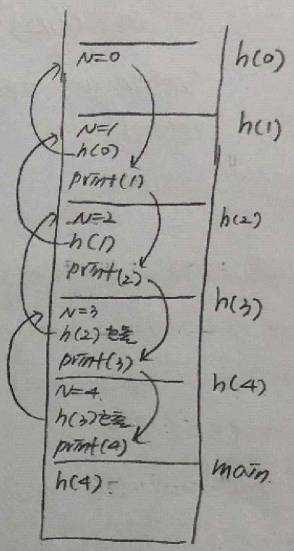
420024.



1. 22 다음의 메소드에 대해 h(4)를 호출한 경우는?

```
01  public static void h(int N) {
02      if (N>0) {
03          h(N-1);
04          System.out.print(N);
05      }
06  }
```

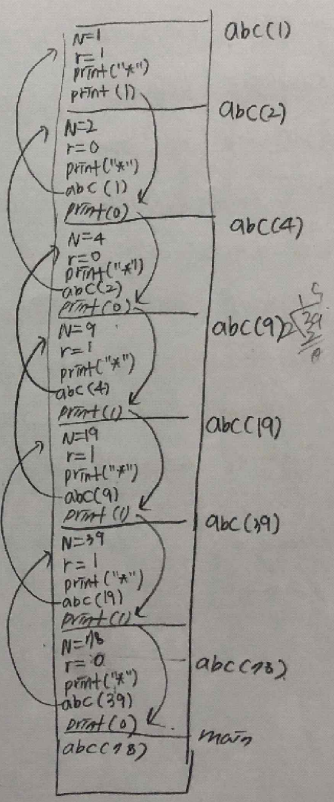
h(4) 호출 → (4>0)이므로 h(3) 호출
→ (3>0)이므로 h(2) 호출
→ (2>0)이므로 h(1) 호출
→ (1>0) 이므로 h(0) 호출
→ (0=0) 이므로 h(0) 종료
→ h(1)에서 1프린트 후 h(1) 종료
→ h(2)에서 2프린트 후 h(2) 종료
→ h(3)에서 3프린트 후 h(3) 종료
→ h(4)에서 4프린트 후 h(4) 종료.




<console>
1234

1. 23 다음의 메소드에 대해 abc(78)을 호출한 결과는?

```
01  public static void abc(int N){
02      int r= N % 2;
03      System.out.print("*");
04      if(N>=2)
05          abc(N/2);
06      System.out.printf("%d", r);
07      return;
08  }
```




abc(78) 호출 → * 프린트 → (78≥2) 이므로 abc(39) 호출

→ * 프린트 → (39≥2) 이므로 abc(19) 호출

→ * 프린트 → (19≥2) 이므로 abc(9) 호출

→ * 프린트 → (9≥2) 이므로 abc(4) 호출

→ * 프린트 → (4≥2) 이므로 abc(2) 호출

→ * 프린트 → (2≥2) 이므로 abc(1) 호출

→ * 프린트 → (1<2) 이므로 1 프린트 후 abc(1) 종료

→ abc(2) 에서 0 프린트 후 종료.

→ abc(4) 에서 0 프린트 후 종료.

→ abc(9) 에서 1 프린트 후 종료.

→ abc(19) 에서 1 프린트 후 종료.

→ abc(39) 에서 1 프린트 후 종료.

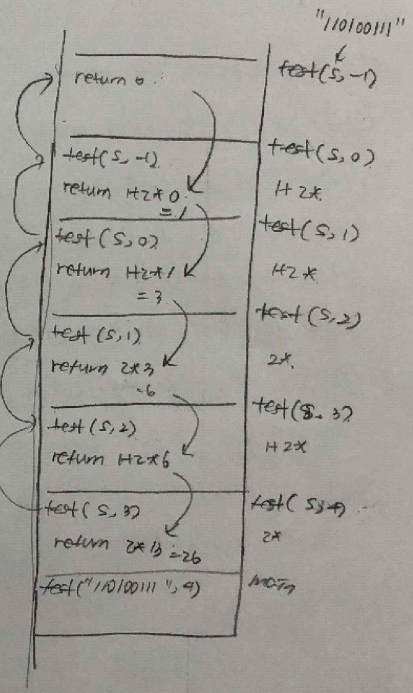
→ abc(78) 에서 0 프린트 후 종료.

<Console>

*******1001110

1.24 다음의 메소드에 대해 test("10100111", 4)가 리턴하는 값은?

```
01  public static int test(String s, int last) {
02      if (last < 0) {
03          return 0;
04      }
05      if (s.charAt(last) == '0') {
06          return 2 * test(s, last-1);
07      }
08      return 1 + 2* test(s, last-1);
09  }
```




test("10100111", 4) 호출 → (4>0).

→ (0==0) 이므로 test("", 3) 호출 → (3>0)

→ (1!=0) test("", 2) 호출 → (2>0)

→ (0==0) 이므로 test("", 1) 호출 → (1>0)

→ (1!=0) test("", 0) 호출 → (0==0)

→ (1!=0) test("", -1) 호출 → (-1<0) 이므로

0리턴 → test(S,0)에서 1리턴

→ test("",1)에서 3리턴

→ test("",2)에서 6리턴

→ test("",3)에서 13리턴

→ test("",4)에서 26리턴.

1.25. 다음의 메소드가 무엇을 계산하는지 설명하시오.

```
01  public static void t(int N) {
02      if(N>0) {
03          t(N/2);
04          System.out.print(N%2);
05      }
06  }
```

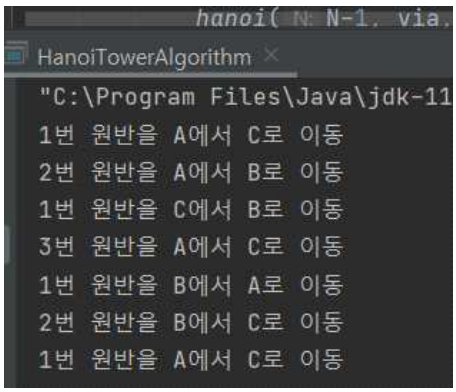
t(6) 0
t(3) 1
t(1) 1
t(0)

예를 들어 t(6)을 호출하였을 때,
t(3), t(1), t(0)이 차례로 호출되고,
t(0)에서 if문의 조건을 만족하지 않음으로
t(1)에서 1%2의 몫과값 1을 프린트,
t(3)에서 3%2의 몫과값 1을 프린트,
t(6)에서 6%2의 몫과값 0을 프린트하여 110이 출력된다.
N을 2로 나누고 그 나머지 값들이 N=0일 때까지 역순대로 보아
원하는 10진수를 2진수로 출력하는 메소드이다.

2|6
2|3 ··0
2|1 ··1
0 ··1

## 프로그램 과제 1 - 하노이탑

(A에 있는 3개의 원반을 모두 C로 옮길때)

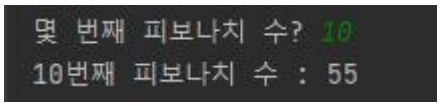


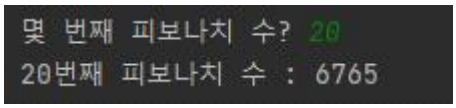

직접 코드를 구현 해보려고 했지만 너무 감이 안잡혀서 결국 하노이탑 알고리즘을 구현하는 과정을 서술한 한 블로그의 글을 봤다. 단순히 코드만 올려놓고 그 코드를 설명하는 것이 아니라 문제소개, 문제 정의, 아이디어 얻기, 재귀, 문제 분해, 실제 코드까지 생각의 흐름과 설계 과정까지 써놓은 글이었다. 처음부터 완성된 코드를 보고 해석하는 반대 과정도 물론 필요하긴 하지만 지금 나의 단계에선 설계를 하고 코딩을 하는 방법을 제대로 배워야 할 것 같다.

설계 과정에서 하노이탑 알고리즘의 재귀적인 특징은 N개의 원반을 옮긴다고 할 때, N-1개의 원반을 옮기는 과정도 발견된다는 것이다. N개의 원반을 모두 옮기려면 N-1번째까지의 원반을 경유지에 옮겨놓는 과정 하나, N번째 원반을 목적지에 옮기는 과정 하나, 또 다시 N-1번째까지의 원반을 목적지로 옮겨놓는 과정 하나, 총 세 번의 과정이 필요하므로 두 번의 hanoi(N-1) 재귀가 필요하다는 것을 알 수 있다.

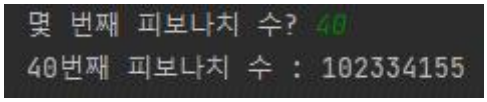
하노이탑이나 피보나치 수열이 아닌 다른 재귀함수 문제를 스스로 풀라고 하면 내가 이렇게 할 수 있을지는 모르겠지만 앞으로는 무작정 키보드에 손을 대고 코드부터 짜는 것이 아니라 최대한 할 수 있을 만큼은 설계를 해보려고 한다.

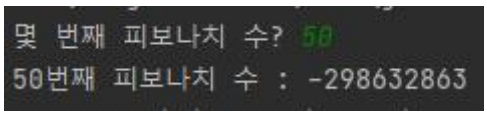
## 프로그램 과제 2 - 피보나치 수열





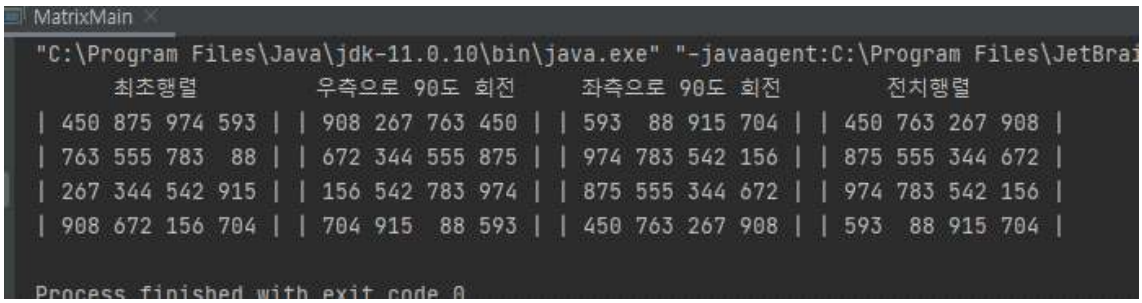



반복함수로 만든 프로그램은 50과 같은 큰 수를 넣어도 결과 값이 바로 나왔던 반면에,
재귀함수로 만든 프로그램은 50을 입력받았을 때 거의 1분이 걸릴 정도로 결과 값이 나오는 속도가 느렸다.
두 프로그램의 수행시간을 비교해보면 O-표기법으로 표현했을 때,
반복문을 사용한 프로그램은 O(N), 재귀함수를 사용한 프로그램은 O(2^N/2)이다.
O(N)과 O(2^N/2)의 수행시간의 차이는 N이 아주 커질 수록 더 커진다. 따라서 반복문으로 작성한 프로그램과 재귀함수를 이용하여 작성한 프로그램의 수행시간 차이는 N이 커질 수록 더 크게 나타난다.

50번째 피보나치 수를 구하려고 했을 때 음수의 값이 나오는데 이는 49번째 피보나치 수를 구할 때부터 int가 저장할 수 있는 범위를 넘었기 때문이다. 이를 오버플로우라 한다.
int가 아닌 long타입으로 바꾸면 50번째 피보나치 수를 알 수 있다.

## 프로그램 과제 3 - 행렬


MatrixMain
"C:\Program Files\Java\jdk-11.0.10\bin\java.exe" "-javaagent:C:\Program Files\JetBrai
최초행렬 우측으로 90도 회전 좌측으로 90도 회전 전치행렬
| 450 875 974 593 | | 908 267 763 450 | | 593 88 915 704 | | 450 763 267 908 |
| 763 555 783 88 | | 672 344 555 875 | | 974 783 542 156 | | 875 555 344 672 |
| 267 344 542 915 | | 156 542 783 974 | | 875 555 344 672 | | 974 783 542 156 |
| 908 672 156 704 | | 704 915 88 593 | | 450 763 267 908 | | 593 88 915 704 |
Process finished with exit code 0