C++

#include <iomanip>

Cout << setprecision(5) << 변수 << ‘\n’; 🡪 유효숫자의 개수가 5개가 되도록 출력

Cout << fixed << setprecision(5) << 변수 << ‘\n’; 🡪 소수점 자리의 개수가 5개가 되도록 출력

* 줄을 바꾸는 경우  
  endl vs \n 🡪 endl의 경우 줄을 바꾼 후 flush의 기능이 추가되어 속도가 느림 (2741번)  
  printf(“$d \n”, i);의 속도가 가장 빠름
* Cin cout을 사용하기 전에 ios\_base::sync\_with\_stdio(false)의 경우 scanner의 속도가 빨라지지만 섞어 사용할 수 없음
* Auto type의 경우 컴파일러가 타입을 추론해서 타입을 결정함 하지만 명확하게 알 수 있어야 합니다.
* 이터레이더  
  for(map<pair<int,int

다이나믹 프로그래밍(다이나믹은 어떤 의미도 없음)  
 - 큰 문제를 작은 문제로 나눠서 푸는 알고리즘  
 1) Overlapping Subproblem(겹치는 부분문제)  
 2) Optimal Substructure(문제의 정답을 작은 문제의 정답에서 알 수 있음)

1로 만들기!(1463)

1. N이 3으로 나눠지면 3으로 나눈다

2. N이 2로 나눠지면 2로 나눈다

3. 1을 뺀다

D[N] = N=>1로 만드는 필요한 연산의 최소값

D[N] = min(D[N/3]+1 , D[N/2]+1 , D[N-1]+1)

2 X n 타일링

2 X n 직사각형을 1 X 2, 2 X 1 타일로 채우는 방법의 수

D[N] = D[N-1] + D[N-2]