**인공지능 과제 1: N queens Problem 보고서**

2019055078 신채영

* 텍스트이(가) 표시된 사진

  자동 생성된 설명Breadth first search, hill climbing, constraint satisfaction problems 세 가지 알고리즘을 이용해 N queens 문제를 해결 (Python3)
* 2019055078\_assignment\_1.py :
  + **BFS: 자료구조 queue를 이용해 모든 경우의 수 탐색** 
    - Board = 각 행에 몇 번째 열에 queen을 배치할지 저장한 배열
    - 텍스트이(가) 표시된 사진

      자동 생성된 설명bfs(board): queue를 이용해 bfs 방식으로 모든 경우를 탐색하고 정의한 bfs\_test로 조건을 만족하는 경우가 있으면 solved=1로 바꾸고 결과를 출력한 후 종료
    - bfs\_test(board): N queens 조건을 만족하는지 확인하는 함수 (가로세로, 대각선)

텍스트이(가) 표시된 사진

자동 생성된 설명

* + **HC(Hill-climbing): 임의의 initial state를 지정하고 h값을 줄이면서 sub-optimal을 찾고 restart를 반복하며 optimal을 찾음**
    - 텍스트이(가) 표시된 사진

      자동 생성된 설명hc(current, n): 정의한 HC함수들을 이용해 current\_state와 queen(새 보드)를 비교하면서 (heuristic비교) hill climbing 진행.
    - 텍스트이(가) 표시된 사진

      자동 생성된 설명HC\_initial\_state(N): Hill climbing의 시작 state를 random하게 지정하고 배열형태로(new\_board) return
    - HC\_count(node, n): 현재 상태의 배열에 서로 attack하는 queen 수(heuristic)를 계산
    - 텍스트이(가) 표시된 사진

      자동 생성된 설명HC\_change(board, n): 현재 위치를 건너뛰고 새로운 위치에 queen을 넣고 새로운 board를 return텍스트이(가) 표시된 사진

      자동 생성된 설명
  + **CSP (forward checking): 모든 variable에 가능한 value를 assign한 상태에서 시작해서 constraint에 맞지 않은 것들을 하나씩 제거하고 empty variable이 생기면 backtracking.** 
    - class Unassigned: unassigned position을 저장하기 쉽게 만든 class
    - 텍스트이(가) 표시된 사진

      자동 생성된 설명csp(grid, queen,n,N): 현재 진행중인 n과 퀸 수 N을 비교해 같으면 답을 출력하고 아닌 경우 다시 forward checking 진행
    - 텍스트이(가) 표시된 사진

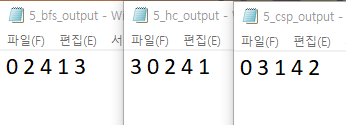
      자동 생성된 설명csp\_change(grid, queen): unassigned position을 result행렬에 모두 넣어 return
    - fc(grid, row, queen): getRowsProposition과 is\_correct를 이용해 현재 남아있는 value들이 규칙을 위반하는지 확인하고 Boolean값 return

텍스트이(가) 표시된 사진

자동 생성된 설명

* + - is\_correct(grid, row, column):
      * rowb(grid, row), columnb(grid, column), upb(grid, row, column), downb(grid, row, column)
    - 텍스트이(가) 표시된 사진

      자동 생성된 설명def getRowsProposition(grid, queen):
  + main:
    - 전역 변수: N(입력), solved(csp board), method(bfs, hc, csp)
    - split으로 N과 method를 입력받고 if문으로 method 실행
      * bfs: 빈 보드 행렬을 bfs함수에 넣고 실행
      * hc: HC\_inital\_state로 시작 state를 만들고 지정한 max restart(100번)만큼 / 답을 찾을 때까지 hill climbing 실행
      * csp: N\*N 행렬을 만들고 (csp\_grid사용) csp함수에 넣고 실행
* 텍스트이(가) 표시된 사진

  자동 생성된 설명**I**nput.txt: (5 queens)
* 결과: 5\_bfs\_output.txt, 5\_hc\_output.txt, 5\_csp\_output.txt