

Artificial Intelligence

Programming Assignment 3 CSP

due : 4/23 일요일 자정 (24:00)

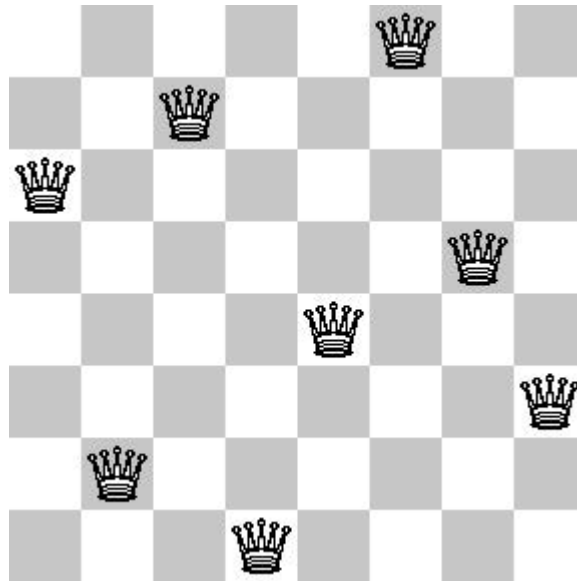
1. 개요

수업시간에 배운 CSP를 활용하여 'N-Queens'문제를 해결한다.

2. 과제 설명

a. 'N-Queens'문제란?

- 'N x N'의 체스판이 있다.
- N개의 Queen이 서로 공격하지 않아야한다.
(Queen은 가로, 세로, 대각선에 있는 말들을 모두 공격할 수 있다.)
- N=8일 때의 답 예시



b. 'N-Queens' 문제 해결

'N-Queens' 문제를 'Standard CSP', 'CSP with Forward Checking', 'CSP with Arc Consistency' 방법을 사용하여 풀어본다. 이 때, 각각의 Search 에서 첫 번째로 찾아진 답을 출력한다. 답을 출력할 때에는, 0~(N-1) 까지의 Column에 있는 각 Queen의 row 위치를 출력한다. (ex. '2-a-iii'의 그림의 답은 2,6,1,7,4,0,3,5 이다.)

c. 입력 형식

모든 입력은 Argument로 받는다. 첫 번째 Argument는 N을 입력받고, 두 번째 Argument는 결과 출력 파일의 '절대 경로'를 입력 받는다. 예시는

아래와 같다.

```
java -jar assignment3.jar 8 D:\Assignment3
```

d. 출력 형식

결과 출력 파일 이름은 “resultN.txt”로 한다. 예를 들어 N=8일 때, 파일 이름은 “result8.txt”이다. 모든 공백은 Space bar 한칸으로 한다.

예) result8.txt

```
>Standard CSP  
Location : 0 4 7 5 2 6 1 3  
Total Elapsed Time : 0.001
```

```
>CSP with Forward Checking  
Location : 0 4 7 5 2 6 1 3  
Total Elapsed Time : 0.0
```

```
>CSP with Arc Consistency  
Location : 0 4 7 5 2 6 1 3  
Total Elapsed Time : 0.001
```

답이 없다면, “Location : ~ “ 대신에 “No solution”을 출력하여 준다.

(assignment 1 참고)

e. 보고서 필수 기재 내용

- i. 본인이 구현한 자료구조 및 알고리즘에 대한 설명
- ii. 약 1~2 분 이내에 측정 가능한 최대 N과 측정 시간에 대한 분석

3. 사용 언어

a. Java

4. 코드 및 보고서 제출 방법

- a. “assignment3” 디렉토리 생성 후 해당 디렉토리 안에 소스코드 위치시킨 후 commit & push 할 것
(과제 3 추가) commit은 한 번만 할 것. 여러번 commit 했을 경우, “git rebase ~~~” 등의 명령어를 통해서 여러 commit을 하나로 합친 후 push 할 것. (GitLab 웹페이지의 각 프로젝트에서 “Repository => Commits”에 과제 3에 대한 commit이 하나만 나타나야 함)
- b. 보고서는 GitLab Wiki에 “assignment3” 페이지 생성 후 작성
- c. GitLab clone 또는 push시에 인증 오류가 난다면, GitLab web page 에서 파일 첨부를 통해 제출 할 것.
- d. GitLab을 통한 자세한 제출 방법은 포털 내 인공지능 강의실 홈페이지 공지 참조

5. 주의 사항

- a. 코드가 완성되면 반드시 GitLab에 push 해야 함
- b. 실행 파일이 아닌, 소스코드를 제출 할 것
- c. 코드에 주석 작성 필수 (주석 없으면 0점 처리)
- d. 컴파일 오류시 0점 처리

- e. 제출 기한 초과시
 - i. 기본 20% 감점 후 매 24시간마다 20%씩 추가 감점
(ex. 50시간 초과시 60% 감점)
 - ii. 72시간 이후 : 0점 처리
- f. 코딩 컨벤션에 따라 코드를 작성하면 추가 점수