

학 기 : 2020년 1학기

과 목 명 : 프로그래밍언어

제출기한 : 2020년 3월 20일 - 04월 10일 (오후 5시)

- 과제 1 -

1. 제목

- 돌 게임(Picking Stones)

2. 목적

- 돌 게임 프로그램 작성을 통해 프로그래밍 알고리즘 대한 이해와 능력을 향상 시킨다.

3. 게임규칙

<ul style="list-style-type: none"><li>1부터 <math>n</math>까지 숫자가 매겨진 <math>n</math>개의 돌로 시작한다.</li><li>Max 플레이어와 Min 플레이어가 번갈아 가면서 돌을 하나씩 가져간다.</li><li>첫 번째 플레이어는 <u><math>n/2</math>보다 작은 숫자</u> 중에서 홀수 돌을 선택해야만 한다. <math>n</math>이 7인 경우(<math>n/2=3.5</math>), 1 과 3 <math>n</math>이 6인 경우(<math>n/2=3</math>), 1</li><li>다음 플레이어는 마지막 제거된 돌의 배수(multiple) 또는 인수(factor)만 가져올 수 있다.(1은 모든 수의 인수)</li><li>플레이어의 차례에 가지고 갈 돌이 없는 경우 패배한다. Max Player : 3 (<math>7/3=3.5</math>이므로 3.5보다 작은 홀수 1또는 3) Min Player : 6 (3의 배수) Max Player : 2 (6의 인수) Min Player : 4 (2의 배수) Max Player : 1 (1은 모든 수의 인수) Min Player : 7 Winner: Min Player (가져갈 돌이 남아 있지 않으므로 Max플레이어 패배)</li></ul>
<ul style="list-style-type: none"><li>가져간 돌이 없으면, Max플레이어가 먼저 시작한다.</li><li>프로그램은 평가 함수를 사용하여 현재 플레이어를 위한 최선의 돌을 선택한다.</li></ul>

4. 구현 내용

구현 내용
<ul style="list-style-type: none"><li>Eclipse(C, C++언어 사용)</li><li>Alpha-Beta pruning algorithm 구현</li><li>입력</li></ul>

play <num\_of\_stones> <num\_of\_taken\_stones> <list\_of\_taken\_stones>

- num\_of\_stones : 게임에 주어진 전체 돌의 개수
- num\_of\_taken\_stones : 가져간 돌의 개수  
0이면 게임 시작이고 Max플레이어 차례  
짝수이면 Max플레이어 차례  
홀수이면 Min플레이어 차례

- list\_of\_taken\_stones : 가져간 돌의 리스트

예) \$ play 7 2 3 6

두 개의 돌(3과 6)을 가지고 갔으므로 Max플레이어 차례

#### ▪ 출력

- alpha-beta 알고리즘으로 계산한 최선의 돌
- alpha-beta 알고리즘으로 계산한 값
- 방문한 노드의 총 숫자
- 노드의 최대 깊이(depth)

예) Input:

\$ play 7 3 1 4 2

#### Output:

Best Move : 6

Calculated Value : 1.0

Number of Visited Nodes : 3

Max Depth : 2

#### ▪ Evaluation

- 게임 종료 시, Max 플레이어 승리 : 1.0
- 게임 종료 시, Min 플레이어 승리 : -1.0
- 그밖에
  - Max 플레이어 차례  
1번 돌을 아직 가지고 가지 않은 경우 : 0반환  
마지막에 1을 가지고 간 경우, 가지고 올 수 있는 숫자 세어서  
홀수면 0.5, 짝수는 -0.5 반환  
마지막에 소수를 가지고 간 경우, 소수의 배수를 계산하여 개수를 세어서  
홀수면 0.7 짝수면 -0.7반환  
마지막에 소수가 아닌 돌을 가지고 간 경우, 소수의 개수를 세어서  
홀수면 0.6 짝수면 -0.6반환
  - Min 플레이어 차례  
1번 돌을 아직 가지고 가지 않은 경우 : 0반환  
마지막에 1을 가지고 간 경우, 가지고 올 수 있는 숫자 세어서

<p>홀수면 -0.5, 짝수는 0.5 반환</p> <p>마지막에 소수를 가지고 간 경우, 소수의 배수를 계산하여 개수를 세어서</p> <p>홀수면 -0.7 짝수면 0.7반환</p> <p>마지막에 소수가 아닌 돌을 가지고 간 경우, 소수의 개수를 세어서</p> <p>홀수면 -0.6 짝수면 0.6반환</p>
<ul style="list-style-type: none"> <li>alpha 초기값 : <math>-\infty</math>(negative infinity)</li> <li>beta 초기값 : <math>\infty</math>(positive_infinity)</li> <li>동일한 값이 계산된 경우, 작은 숫자의 돌을 선택한다. 예) 2와 6의 평가 값이 같은 경우, 2를 선택한다</li> </ul>

## 5. 채점 기준 (10점)

<p>보고서(10점)</p> <ul style="list-style-type: none"> <li>Alpha-Beta pruning algorithm 설명(5점)</li> <li>구현 자료구조 및 함수 설명(5점)</li> </ul> <p>프로그램(10점)</p> <ul style="list-style-type: none"> <li>돌 게임 실행(5점)</li> <li>Alpha-Beta pruning algorithm 구현 (5점)</li> </ul> <p>※ 보고서 프로그램 개별 채점</p> <ul style="list-style-type: none"> <li>구현 안한 경우 설계한 자료구조 및 함수 설명하여 소스코드 없이 보고서만 제출</li> </ul>
---

## 7. 문서 제출

<ul style="list-style-type: none"> <li>소스코드(프로젝트파일)와 보고서를 “학과_이름_hw1.zip” 형식으로 압축하여 vanillo97@gmail.com 메일로 제출. - 프로젝트 폴더 전체를 압축하여 제출.</li> <li>주의사항 <ul style="list-style-type: none"> <li>과제 마감 당일에는 과제 제출자가 많아서 과제제출이 정상적으로 되지 않을 수 있으므로, 되도록이면 미리 제출 바랍니다.</li> <li>과제 채점 시 카피본이 발견되면 카피한 학생과 제공자 모두 한 학기 과제 점수 0 점 처리. 참조한 문헌이나 사이트가 있다면 보고서에 표기해주세요.</li> <li>마감 시간이 경과 후 제출한 경우, 감점 10점</li> </ul> </li> </ul>
---