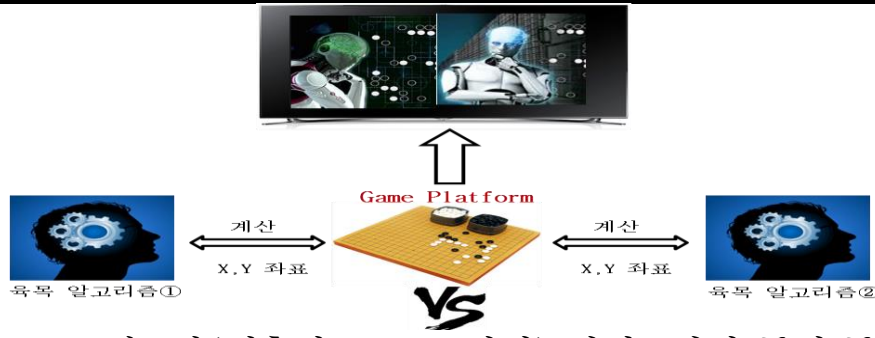


육목 알고리즘 대회 개발 Guide

- 내용 : Game AI 개발 통한 알고리즘 대회
 - 운영 : 조별 리그 및 팀 토너먼트
 - 게임 규칙 :
 - **최초 1회 1수 후, 2수씩 놓음**
 - 시작은 흑이 위치에 상관없이 1개를 놓을 수 있으며,
그 후 백부터 게임 종료까지 2개씩 원하는 위치에 놓을 수 있습니다.
 - **6개 연속된 돌을 직선/대각선으로 만들면 승리** ※육목 초과 시 실격패 (7목 이상)
 - 게임 승리조건은 6목만 이며, 초과되는 7목, 8목 등은 실격패 입니다.
 - **제한시간 초과 시 실격패 (2~7 초 Random / 1경기)**
 - 1 Turn에 대한 제한시간 입니다. 두수 계산시에도 제한시간 내 계산해야 함
 - 1:1 경기 진행 중 제한시간 초과 시 상대편 승리 (무응답, 시간 초과 등)
 - **최대 메모리 사용(6GB) 초과 시 실격패**
 - **바둑판에 돌을 놓을 수 없는 곳이 랜덤으로 존재함(Blocking)**
 - 최대 10개 이내의 짝수 개로 존재, Blocking 자리에 놓을 시 실격패
 - Blocking은 흑/백 양쪽 돌로 인정 (자신의 돌로 활용 가능함)
 - **무승부 경우 제한조건 변경 (Block↑)**
 - 무승부의 경우 제한조건 변경 후 재경기
 - 그래도 승패 결정이 어렵다면 주최측의 AI와 경기 통해 승자를 결정함
(소모 Turn을 고려하여 승자 결정)
- ※ 제한조건은 경기마다 Random 변경됨(3판 2승제의 경우 3판 모두 제한조건 달라질 수 있음)
- Ground Rule
 - 제출한 Source 빌드 불가 시 실격패
 - 게임 Platform에서 실행되지 않으면 실격패
 - 자신의 Turn 종료 후 다른 기타 작업 불가(BackGround 작업 X)
 - 유사도 체크를 통해 비율 높을 시 실격패 (오픈소스 활용금지)
- ※ 재경기 관련
 - 플랫폼 오류 / 외부 기타 이슈로 경기 진행 불가 시, 주최측 판단하여 재경기
 - 지원자 AI의 문제로 재경기는 불가함 (무응답, 시간초과, 오류 등)
- ※ 제출 Source가 본인의 Source가 아닌 경우 지원자격 취소 예정
 - 외부 공개된 Source 코드를 그대로 사용하는 경우 (유사도 25% 이상)
 - 타인이 대신 Source 코드를 작성한 경우 (다른 팀 Source 활용 등)

(우승팀은 부정행위 발견 시 수상내역 취소되며 상금 전액 반환함)

2. 개발 Guide

구 분	내 용
시스템 구성도	 <p>- 프로세스간(제출된 AI, exe파일) 파이프라인 통해 통신</p>
사용가능 언어	- C, C++
실행환경	<ul style="list-style-type: none"> - O/S : Windows 7 64bit - CPU : i7-6700 3.40GHz / NVIDIA Geforce GTX1050 - Ram : 16GB (변동가능성 있음, 유사환경)
빌드환경	<ul style="list-style-type: none"> - Visual Studio 2013(권장) / 2015 ※ VC++ 2013 컴파일러로 빌드 예정
제 공	<ul style="list-style-type: none"> - 육목 게임 Platform (게임 및 AI Test 가능) - 예제파일 (개발 Source ↔ Platform 통신)
API 정보	<ul style="list-style-type: none"> - int showBoard(int x, int y) : 바둑판 현황 [0: 빈공간, 1: 내가 둔 돌, 2: 상대방 돌, 3: Blocking] - void domymove(int x[2], int y[2], int cnt) [바둑돌 놓기, X/Y좌표, cnt- 놓아야 될 바둑돌 수]
제약조건	<ul style="list-style-type: none"> - 네트워크 사용불가(local PC자원만 활용) - 제한시간 초과 시, 프로세스 강제 종료 - 메모리 사용률 초과시, 실격패 (6GB) - 19 X 19 바둑판 /자신의 Turn만 활용(Background 작업X) - 유사도 높은 Source 실격처리
결과물 제출	<ul style="list-style-type: none"> - 기한 內 Email 제출 (ds.sw@samsung.com) - 메일제목에 “[육목제출]” 반드시 포함 - 제출 파일 ※제출된 Source 정상 빌드/실행되지 않을 시, 실격 . Connect6Algo.h /cpp , AI_Algorithm_Code.cpp ※ 추가파일 없이 위의 파일만 제출, 기존 빌드환경에 위의 파일만 대체하여 빌드 예정

#별첨1. Game Platform 화면구성 (육목)

#별첨2. 개발 참고사항

- 이 상 -

#별첨1. Game Platform 화면구성(육목)

구 분	내 용
Game Platform 화면 구성	① 메인화면(Menu)
	 <p>The main menu screen features a dark, sci-fi background with a large spaceship. In the center, there are five buttons: '연습하기' (Practice), '토너먼트' (Tournament), '환경설정' (Settings), '도움말' (Help), and '종료' (Exit).</p>
	② 연습하기(AI/사람 Test 가능)
	 <p>The practice screen shows a Go board on the left and a match setup area on the right. The match setup area includes the 'ALGO CUP' logo, '알고리즘배틀' (Algorithm Battle), 'Player1 VS Player2', a timer set to 05:00, and buttons for '시작' (Start) and '뒤로' (Back).</p>
	③ AI 선택 (바둑돌 클릭)
	 <p>This screen shows the AI selection process. A file explorer window is open, displaying a list of AI files. A red arrow points to a file named 'Player2' in the match setup area, with the label '클릭' (Click) next to it. The '시작' (Start) and '뒤로' (Back) buttons are at the bottom.</p>
	④ 환경설정 (Blocking, 시간제한 설정)
	 <p>The settings screen is divided into several panels: '모니터' (Monitor) with a dropdown menu, '규칙' (Rules) with checkboxes for '블로킹' (Blocking) and '시간제한' (Time Limit), '블록' (Block) with a slider, '게임' (Game) with a dropdown for '토너먼트 선승' (Tournament Win), '커스터마이즈' (Customize) with dropdowns for '바둑돌 커스터마이징' (Go Stone Customization) and '알판바둑판' (Go Board), and '리그모드' (League Mode) with radio buttons for '리그 모드(Normal)' and '리그 모드(Only)'.</p>

#별첨2. 개발참고 사항

1. 배포 파일

- ① Game Platform 설치파일 : Setup.zip (Setup.exe, Setup.msi)
- ② 예제파일 : SamsungGo_AI_Code_예제파일.zip

2. 프로토콜 (Standard Input/Output 통한 통신)

- ① Input : START
Output : OK
내용 : 바둑판 초기화
- ② Input : INFO
Output : char info[] = { "TeamName:0000,Department:0000" };
내용 : 팀명, 소속학교
0000 이부분만 수정 / 10글자 내외
- ③ Input : BLOCK X,Y
Output : OK
내용 : 돌을 놓을 수 없는 위치 정보 (Blocking 정보)
- ④ Input : BEGIN
Output : X,Y
내용 : 게임의 첫수 놓기
- ⑤ Input : TURN X,Y 또는 TURN X1,Y1 X2,Y2
Output : X1,Y1 X2,Y2
내용 : 입력값 TURN의 위치좌표는 상대방 바둑돌 놓은 정보
반환값으로 내가 놓을 돌의 위치정보
- ⑥ Input : QUIT
Output : 없음
내용 : 프로그램 종료
- ⑦ Input : LimitTime 0 -> 제한시간
Output : 없음
내용 : 제한시간 (2~7초 Random)

※ fflush(std) 호출 필요함

※ FAQ

- 욱목 초과, 즉 7목도 인정되는 건가요?
 - 욱목만 인정되며 7목 이상 시 실격패 입니다.
- 반드시 제공되는 파일을 사용해야 하나요?
 - 기본 제공되는 파일만 사용! 단, 변수/함수 추가는 가능함
(AI_Algorithm_Code.cpp / Connect6Algo.h / Connect6Algo.cpp 파일만 변경)
- 제한된 라이브러리나 오픈소스 사용등 불가능한 것들이 있나요?
 - 외부 자원과 지원자 본인이 구현하지 않은 외부 라이브러리 사용은 불가입니다.
 - 유사도 체크 통해 유사도 높은 팀은 실격처리 예정이라, 오픈소스 사용 시 주의필요
 - 유사도 25~30% 이상의 소스 코드 발견 시, 욱안으로 대조 검증 예정
※ 유사도 체크는 Stanford MOSS를 기본으로 1~2가지 Tool 더하여 검증 예정
- C,C++ 외 다른언어는 사용하면 안되나요?
 - C,C++ 두개 언어만 사용하여 대회 참여해주시기 바랍니다.
- background로 지속 Thread생성해서 동작해도 무방할까요?(상대방Turn)
 - 자신의 Turn 만 활용해주시기 바랍니다.
바둑돌을 놓은 후 별도의 Background 작업 통해 자원사용 X
- 상대방 직전 수
 - 매번 확인할 필요는 없고, Opmove 함수에서 상대방 돌을 Board에 갱신하고 있습니다.
변수 선언하셔서 해당 위치에 마지막 좌표값을 가지도록 하면 매번 확인할 필요 없음
- 로그는 어떻게 확인하나요?
 - 플랫폼 로그는 설치된 경로의 log.txt 파일 확인하시면 됩니다.
 - 개발하시는 프로세스에서 확인이 필요하시면, 파일 입출력을 활용하셔서 로그를 남기셔서 작업하시면 수월하시니 참고 해주시기 바랍니다.
(제공된 예제파일 참고, myLog.txt파일에 남기고 있음)
- 기계학습 알고리즘을 사용 가능한가요?
 - 사용하는 알고리즘이나 개발 방법에 별도의 제약은 없습니다.
 - 따라서 기계학습 프레임워크를 사용하여 개발하는 것도 가능합니다.
 - 단, 최종 제출하는 코드는 외부 의존성 없이 빌드 및 실행이 가능해야 합니다.
(경기용 PC에 추가적인 라이브러리/프레임워크/dll 파일 등 설치 불가)

※ 경기용 PC에 텐서플로 파이썬 3.5 CPU 버전으로 설치 예정

인터페이스, 학습결과물 모두 코드상에 본인이 구현하여 해당파일만 제출

※ Python Tensorflow 활용할 경우 아래 사항 참고해주세요. (C,C++만 사용가능함)

C++에서 해당 Python 스크립트를 실행할 수 있도록 C++ Boost 라이브러리를 활용하여 개발하실 것을 권장합니다. Tensorflow의 경우 Python으로 개발한 뒤, C++에 Python을

Embedding하여 소스를 작성하는 것은 가능합니다. (Boost 라이브러리 참고)

파이썬 홈 디렉토리 경로- C:\Anaconda3\envs\py36

- 구현한 exe Platform에서 실행시 "올바르지 않은 AI파일 입니다" ,
"INFO가 잘못되었습니다" 의 메시지가 나와요

- 게임 플랫폼을 관리자 권한으로 실행
- 해당 문제는 Visual C++ 버전이 일치하지 않아 발생하는 문제로, VS 재배포 가능 패키지 설치를 통해 해결할 수 있습니다.
 - <https://www.microsoft.com/ko-kr/download/details.aspx?id=48145> (VS2015)
 - <https://www.microsoft.com/ko-KR/download/details.aspx?id=40784> (VS2013)