**Artificial Intelligence Battle Sheep Spring 2022 Group Game Project Spec**

**About programming**

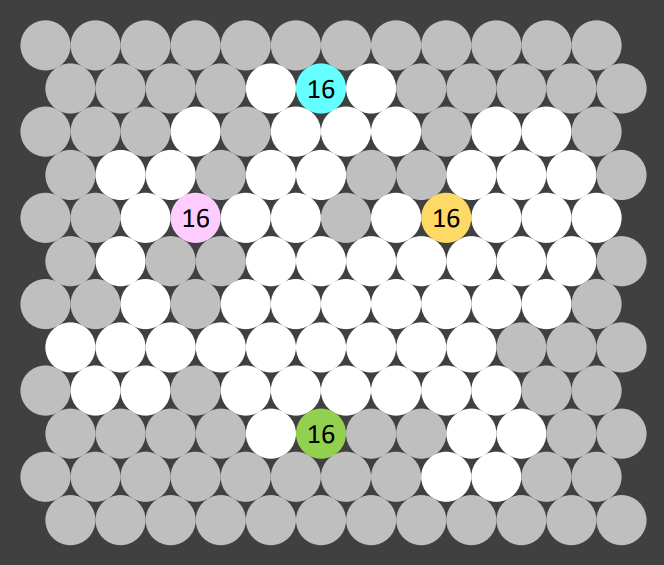
You are allowed to use Python (3 or above)/C++ (14 or below) to code your program. TAs will provide code including TCP socket and a basic code template. For simplicity, you only need to fill in the function provided in template code.

* The main program **AI\_game.exe** takes the content of “input.txt” as input. Change the text in this file to run your code.
* General

you don’t need to worry about the connection between your program and the tournament judge if you follow the template code. The template code provides a function **InitPos** and **GetStep** for you to fill in and the returned object contains the current action made by you program

* Parameter Explain
  + **playerID**: your role in this game (1~4)
  + **mapStat**: the state of each player’s domain which is a 12\*12 array with number -1 to 4. In this array, 0 infers to free area, -1 means there is a wall, 1~4 shows that place occupied by player 1~4.
  + **sheepStat**: # sheep at each place. It is a 12\*12 array and range from 0 to 16.
  + **idTeam** inSTcpClient.cpp/.py: it should be change to your group number.

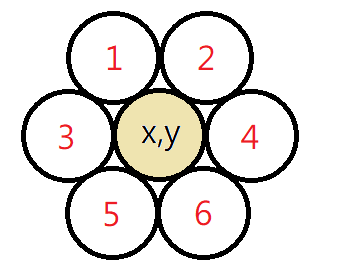
x, y in mapStat and sheepStat is defined as below.



**(x,y)=(4,6)**

* Function **InitPos**
  + It will be called after you receive initial map from server. What you need to do is to decide where to start the game.
  + InitPos returns init\_pos, which is infer to the place you choose as start place. Its format is (x,y).
* Function **GetStep**
  + It will be called only when it’s your turn (no matter what your player role is) and only when there are still some valid moves in your turn. That is, you don’t need to worry about the situation when there is no valid move.
  + GetStep returns 3 information:
    - select position (x, y)
    - #sheep moving to new place
    - moving direction (1~6). Direction code is showed below.

Its format is a tuple in python: [(x, y), #sheep, moving direction] and showed as std::vector<int> with size 4 in C++.



* Python
  + **STcpClient.py**:   
    This code includes TCP functions. You should not change it except that you must change the number at line 13 “idTeam = -1” into your team number “idTeam = ”. We will announce the team number later.
  + **Sample.py**:   
    This is template code which includes function InitPos and GetStep to be filled by you.
  + We will run your code in python 3.6 or above with win10 operating system. Your source code should be able to run directly by python interpreter.
* C++
  + **STcpClient.h**:   
    This code includes TCP functions. You should not change it except that you must change the number at line 18 “idTeam = -1” into your team number “idTeam = ”. We will announce the team number later.
  + **Sample.cpp**:   
    This is template code which includes function InitPos and GetStep to be filled by you.
  + Notice that template code is written in Visual Studio 2019 with win10 operating system and includes OS-depended code (winsock for TCP) and compiler depended code (#pragma). Both compiler and operating system can be found on <https://ca.nctu.edu.tw/>.
  + We will run your code in win10 operating system. DO NOT submit the whole project directory. Instead, just submit STcpClient.h and Sample.cpp.

**TAs strongly recommend you using the same compiler and operating system as mentioned above to ensure there won’t be any unexpected problems**