Instructions

Проекты для роботов сгруппированы в подкаталог Robots, для начала работы со своим роботом нужно взять проект из каталога Robots\robotbase.NN, где NN – номер по журналу.

Для включения работающего робота в турнир необходимо добавить строчку с именем библиотеки в файл robottournament\list.txt.

Поведение робота описывается в функции DoStep.

Функция, вызываемая ядром, должна иметь следующий вид:

void DoStep(stepinfo \*Info, step \*Step)

Для совершения действия вызывайте функцию DoAction:

inline void DoAction(step \*Step, int type, int dx, int dy)

inline void DoAction(step \*Step, int type, int A, int P, int V)

inline void DoAction(step \*Step, int type, int dx = 0, int dy = 0, int A = 0, int P = 0, int V = 0)

1. Первым параметром передавайте Step, это структура, в которую пишется действие.
2. type – тип действия, которое вы хотите совершить. Варианты:

ACT\_MOVE

ACT\_ATTACK

ACT\_TECH

Обратите внимание, что действия производятся с учетом параметров и положений на начало хода, т.е. атака, например, будет совершена с силой, заданной в прошлом ходу.

1. dx и dy – координаты относительно ваших, на которые будет направлено действие (перемещение ACT\_MOVE или атака ACT\_ATTACK).
2. A, P, V – новые значения параметров при распределении (ACT\_TECH).

Функция имеет перегрузки, поэтому:

* при перемещении и атаке передавайте Step, ACT\_MOVE или ACT\_ATTACK, dx, dy
* при перераспределении параметров передавайте Step, ACT\_TECH, A, P, V
* или честно передавайте Step, тип действия и все параметры, в т.ч. те, что при данном действии не используются (например: DoAction(Step, ACT\_TECH, 0,0,newA,newP,newV).

Информация о поле содержится в структуре Info.

struct stepinfo

{

int stepnum;

int yourNumber;

robotinfo \*\*robots;

object \*\*objects;

step \*\*history;

fieldData field;

};

stepnum – номер хода

yourNumber – Ваш номер в списке роботов

robots – массив с текущими данными всех роботов

playerid – идентификатор игрока, у роботов, принадлежащих одному игроку, идентификатор совпадает.

objects – массив с текущими данными всех объектов

type – тип объекта: OBJ\_CHARGER – электростанция, OBJ\_TECH – тех. станция.

history – данные о предыдущем шаге для каждого робота (индексы те же, что в robots)

field – информация о параметрах поля

Данные, передаваемые в каждом поле, можете узнать, посмотрев определения соответствующих структур в файле Parameters.h.