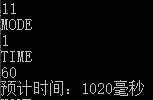
棋盘将121个格子从左到右从上到下定义为了0-120号，根据其ID确定具体位置

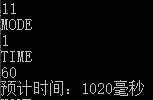
首先必须输入棋盘大小：N

输出：MODE，界面用以判断行为



{

当AI先手（AI执红）时：



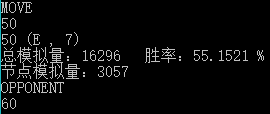
输入：1，用以判别先后手

输出：TIME，界面用以判断行为

输入运行时间：一个数，根据引擎确定

输出：预计时间：多少多少秒

继续输出：



MOVE（一个标志，界面用以判别引擎的行为）

落子的位置：0-120

落子的位置在棋盘上的表示：0-120 （A-K，1-11），例如 50 （E，7）//可以没有

总模拟量：12345 胜率：89% //可以没有

节点模拟量：12121 //可以没有

OPPONENT（结束标志，界面用以判别是否停止输出）

循环

{

输入：落子位置

输出：TIME

输入：运行时间参数

输出：

MOVE（一个标志，界面用以判别引擎的行为）

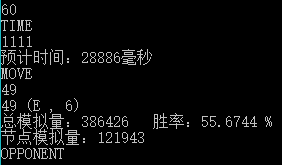
落子的位置：0-120

落子的位置在棋盘上的表示：0-120 （A-K，1-11），例如 50 （E，7）//可以没有

总模拟量：12345 胜率：89% //可以没有

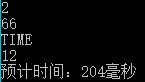
节点模拟量：12121 //可以没有

OPPONENT



}

当人先手（AI执蓝）时：



输入：2用以判别先后手

此时无输出

输入：落子位置

输出：TIME，界面用以判断行为

输入运行时间：一个数，根据引擎确定

输出：预计时间：多少多少秒

继续输出：

MOVE（一个标志，界面用以判别引擎的行为）

落子的位置：0-120

落子的位置在棋盘上的表示：0-120 （A-K，1-11），例如 50 （E，7）//可以没有

总模拟量：12345 胜率：89% //可以没有

节点模拟量：12121 //可以没有

OPPONENT

循环

{

输入：落子位置

输出：TIME

输入：运行时间参数

输出：

MOVE（一个标志，界面用以判别引擎的行为）

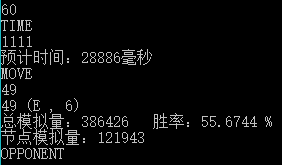
落子的位置：0-120

落子的位置在棋盘上的表示：0-120 （A-K，1-11），例如 50 （E，7）//可以没有

总模拟量：12345 胜率：89% //可以没有

节点模拟量：12121 //可以没有

OPPONENT



}

}