greedy(単純なスコアマップを取り入れよう編)

　田中 健人

greedyに単純なスコアマップを取り入れる。

ソースは以下。このAIをgreedy\_map1と呼ぶ。

<https://drive.google.com/file/d/1yuqEJBJOO8IcBNzyZC888CuvtCSoLvU7/view?usp=sharing>

greedy\_map1はゴール+視野の列を0として、そこから、pseudo\_alphabetaで使われているスコアマップと同じように、スタートまでスコアを付ける。

今回、ゴール+視野からスコアマップを作った理由は、こうすることで、視野外では、y軸を進めるほどスコアが良く、視野内では障害物を考慮してスコアが設定されるということと、今後視野外の情報も敵の動きでスコアを設定する目論見があるため、拡張しやすいようにそうした。

このAIをサンプル4種類+pseudo\_alphabeta\_cal3と戦わせる。

公式のルールと同じで、1ゲームplayerを交代し2レースで構成。

2レースどちらも勝てば勝ち、片方勝ち、片方負ければあいこ。

11種類のコース全て、11ゲーム行い、より多くゲームで勝利したほうが、最終的な勝ちとする。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | greedy | LookaheadPlayer | pseudo\_alphabeta | py3\_sample | pseudo\_alphabeta\_cal3 |
| greedy\_map1 | ×  4-7 | 〇 9-2 | 〇  7-4 | 〇  9-2 | 〇  6-5 |

以下、各勝負の詳細を記述する。

(先行) VS (後攻)であり、先行は第一レースでplayer1、第二レースでplayer2

スコアは小数点第二で四捨五入。

(player1) - (player2)で表示。

先行視点で、勝ち負けを表記。

<greedy\_map1 VS greedy>

* sample(負け)
  + 第一レース  
    55.2 - 52.8で負け。
  + 第二レース  
    54 - 53.2で勝ち。
* 01(負け)
  + 第一レース  
    30.2 - 28.7で負け。
  + 第二レース  
    29.7 - 29.2で勝ち。
* 02(負け)
  + 第一レース  
    47.2 - 42.4で負け。
  + 第二レース  
    44.4 - 43.2で勝ち。
* 03(負け)
  + 第一レース  
    77.3 - 78.5で勝ち。
  + 第二レース  
    80 - 81.3で負け。
* 04(勝ち)
  + 第一レース  
    68.2 - 74.4で勝ち。
  + 第二レース  
    70.4 - 72.2で負け。
* 05(勝ち)
  + 第一レース  
    25 - 220で勝ち。
  + 第二レース  
    13 - 28で負け。←偶然ゴールされちゃった感じです。
* 06(負け)
  + 第一レース  
    47.3 - 43.8で負け。
  + 第二レース  
    45.5 - 47.2で負け。
* 07(勝ち)
  + 第一レース  
    37.2 - 38.3で勝ち。
  + 第二レース  
    41.4 - 39.1で勝ち。
* 08(勝ち)
  + 第一レース  
    44.2 - 43で負け。
  + 第二レース  
    44.6 - 42.3で勝ち。
* 09(負け)
  + 第一レース  
    19.3 - 19で負け。
  + 第二レース  
    19 - 20.3で負け。
* 10(負け)
  + 第一レース  
    36.2 - 31.2で負け。
  + 第二レース  
    35.7 - 39.2で負け。

<greedy\_map1 VS LookaheadPlayer>

* sample(負け)
  + 第一レース  
    53.2 - 44.8で負け。
  + 第二レース  
    54.4 - 52.2で勝ち。
* 01(勝ち)
  + 第一レース  
    30.2 - 32.7で勝ち。
  + 第二レース  
    36 - 29.2で勝ち。
* 02(勝ち)
  + 第一レース  
    42.2 - 51.5で勝ち。
  + 第二レース  
    50.5 - 42.2で勝ち。
* 03(勝ち)
  + 第一レース  
    77.3 - 80.7で勝ち。
  + 第二レース  
    81.7 - 78.3で勝ち。
* 04(勝ち)
  + 第一レース  
    68.2 - 76で勝ち。
  + 第二レース  
    300 - 72.2で勝ち。
* 05(負け)
  + 第一レース  
    220 - 18で負け。
  + 第二レース  
    18 - 23で負け。
* 06(勝ち)
  + 第一レース  
    47.2 - 200で勝ち。
  + 第二レース  
    200 - 47.2で勝ち。
* 07(勝ち)
  + 第一レース  
    38.1 - 39.7で勝ち。
  + 第二レース  
    38.2 - 39.4で負け。
* 08(勝ち)
  + 第一レース  
    43.3 - 49で勝ち。
  + 第二レース  
    43 - 41.3で勝ち。
* 09(勝ち)
  + 第一レース  
    19.3 - 21で勝ち。
  + 第二レース  
    19 - 20.3で負け。
* 10(勝ち)
  + 第一レース  
    32.2 - 34.3で勝ち。
  + 第二レース

41.2 - 31.3で勝ち。

<greedy\_map1 VS pseudo\_alphabeta>

* sample(勝ち)
  + 第一レース  
    53.2 - 59.5で勝ち。
  + 第二レース  
    52.8 - 53.2で負け。
* 01(勝ち)
  + 第一レース  
    31.2 - 120で勝ち。
  + 第二レース  
    120 - 22.2で勝ち。
* 02(勝ち)
  + 第一レース  
    42.2 - 200で勝ち。
  + 第二レース  
    200 - 42.2で勝ち。
* 03(勝ち)
  + 第一レース  
    77.3 - 380で勝ち。
  + 第二レース  
    380 - 80.3で勝ち。
* 04(勝ち)
  + 第一レース  
    68.2 - 300で勝ち。
  + 第二レース  
    300 - 72.2で勝ち。
* 05(負け)
  + 第一レース  
    220 - 50で負け。
  + 第二レース  
    50 - 23で勝ち。
* 06(負け)
  + 第一レース  
    47.2 - 41.8で負け。
  + 第二レース  
    41.8 - 47.2で負け。
* 07(負け)
  + 第一レース  
    38.2 - 34.3で負け。
  + 第二レース  
    40.3 - 39.2で勝ち。
* 08(負け)
  + 第一レース  
    44.3 - 41.8で負け。
  + 第二レース  
    36.8 - 42.3で負け。
* 09(勝ち)
  + 第一レース  
    19.3 - 100で勝ち。
  + 第二レース  
    19.3 - 100で勝ち。
* 10(勝ち)
  + 第一レース  
    26.1 - 29.4で勝ち。
  + 第二レース

36.8 - 36.2で勝ち。

<greedy\_map1 VS py3\_sample>

* sample(負け)
  + 第一レース  
    52.2 - 48.5で負け。
  + 第二レース  
    53.5 - 54.2で負け。
* 01(勝ち)
  + 第一レース  
    30.2 - 34で勝ち。
  + 第二レース  
    33.8 - 29.2で勝ち
* 02(勝ち)
  + 第一レース  
    42.2 - 200で勝ち。
  + 第二レース  
    200 - 42.2で勝ち。
* 03(勝ち)
  + 第一レース  
    77.3 - 380で勝ち。
  + 第二レース  
    380 - 80.3で勝ち。
* 04(勝ち)
  + 第一レース  
    68.2 - 300で勝ち。
  + 第二レース  
    300 - 72.2で勝ち。
* 05(負け)
  + 第一レース  
    220 - 35で負け。
  + 第二レース  
    36 - 23で勝ち。
* 06(勝ち)
  + 第一レース  
    50.2 - 53で勝ち。
  + 第二レース  
    53 - 48.2で勝ち。
* 07(勝ち)
  + 第一レース  
    37.1 - 42で勝ち。
  + 第二レース  
    44 - 39.1で勝ち。
* 08(勝ち)
  + 第一レース  
    43.3 - 48で勝ち。
  + 第二レース  
    44 - 45.3で負け。
* 09(勝ち)
  + 第一レース  
    19.3 - 100で勝ち。
  + 第二レース  
    100 - 20.3で勝ち。
* 10(勝ち)
  + 第一レース  
    31.3 - 38で勝ち。
  + 第二レース

35 - 33.2で勝ち。

<greedy\_map1 VS pseudo\_alphabeta\_cal3>

* sample(負け)
  + 第一レース  
    53.2 - 45で負け。
  + 第二レース  
    50.5 - 51.2で負け。
* 01(勝ち)
  + 第一レース  
    31.2 - 120で勝ち。
  + 第二レース  
    120 - 29.2で勝ち。
* 02(勝ち)
  + 第一レース  
    42.2 - 200で勝ち。
  + 第二レース  
    200 - 42.2で勝ち。
* 03(勝ち)
  + 第一レース  
    77.3 - 380で勝ち。
  + 第二レース  
    380 - 78.3で勝ち。
* 04(勝ち)
  + 第一レース  
    68.2 - 300で勝ち。
  + 第二レース  
    300 - 71.2で勝ち。
* 05(負け)
  + 第一レース  
    220 - 50で負け。
  + 第二レース  
    50 - 23で勝ち。
* 06(負け)
  + 第一レース  
    47.2 - 41.8で負け。
  + 第二レース  
    41.8 - 47.2で負け。
* 07(負け)
  + 第一レース  
    39.2 - 34.3で負け。
  + 第二レース  
    33.3 - 39.2で負け。
* 08(負け)
  + 第一レース  
    42.3 - 39.8で負け。
  + 第二レース  
    36.8 - 42.3で負け。
* 09(勝ち)
  + 第一レース  
    19.3 - 100で勝ち。
  + 第二レース  
    100 - 19.3で勝ち。
* 10(勝ち)
  + 第一レース  
    26.1 - 28.8で勝ち。
  + 第二レース

28.8 - 31.2で負け。

いや、greedyに負けてどうすんねん。

確かに負けたけど、仕方ないと思うんよな....障害物を避けて通る分スピード落ちて、

ただただ先に進むgreedyに負けるのは。

でも、ステージのこととか考えて、それじゃだめだし、greedy\_map1のようにせざるを得ないしな。

まあ、greedyについて、まだまだ考えないかんことはたくさんあるから、今後のタスクで改良を...