

# 第 37 回 全国高等専門学校 プログラミングコンテスト 競技部門「ヘキサうどん」のフォーマットについて

## 試合開始前のマップ構成フォーマット

- ・ 試合開始前に、マップ構成を次のような JSON 形式データで提供します。
- ・ 試合時間に関する情報は、"startsAt" キーに試合の開始時刻(UNIX TIME)を記録します。
- ・ "daySeconds"キーに試合内での 1 日が経過するまでの秒数を、"daySteps" キーに各日のステップ数を記録します。"daySteps" キーの配列の要素数が日数を表します。
- ・ "map" キーにマップの地形情報を、"spots" キーにスポットの情報を記録します。
- ・ 地形情報では、"height"キーがマップの縦幅、"width"キーがマップの横幅、"cells"キーがマップの各セルの地形タイプ(0：平地，1：道路，2：山地，3：池)を表します。
- ・ スポットの情報では、"brand"キーがスポットの系列、"pos"キーがスポットの位置、"stocks"キーが最大在庫数を表します。
- ・ エージェントに関する情報は、"agents" キーに自プレイヤーのエージェント初期位置を記録します。試合中はリスト内の順序がエージェントに割り当てられ、後述の試合情報フォーマット、エージェント種別の回答フォーマット、行動計画の回答フォーマットも同じ順序で記述します。
- ・ "fuelLimits" キーに燃料積載量上限を記録します。
- ・ 交通量に関する情報は、"players" キーに参加プレイヤー数を、"busyThreshold" キーに混雑基準値を、"jammedThreshold" キーに渋滞基準値を記録します。

```
{
  "startsAt": 1778227200,
  "daySeconds": 5,
  "daySteps": [50, 100, 150, 200],
  "map": {
    "height": 8,
    "width": 8,
    "cells": [
      [3, 0, 1, 2, 0, 1, 2, 0],
      [3, 0, 1, 2, 0, 1, 2, 0],
      [3, 0, 1, 2, 0, 1, 2, 0],
      [3, 0, 1, 2, 0, 1, 2, 0],
      [3, 0, 1, 2, 0, 1, 2, 0],
      [3, 0, 1, 2, 0, 1, 2, 0],
      [3, 0, 1, 2, 0, 1, 2, 0],
      [3, 0, 1, 2, 0, 1, 2, 0]
    ]
  },
  "spots": [
    {"brand": 0, "pos": 1, "stocks": 4},
    {"brand": 1, "pos": 9, "stocks": 1},
    {"brand": 0, "pos": 17, "stocks": 1},
    {"brand": 1, "pos": 25, "stocks": 3}
  ],
  "agents": [4, 12, 20, 28],
  "fuelLimits": 20,
  "players": 8,
  "busyThreshold": 2,
  "jammedThreshold": 4
}
```

## 各日開始時の試合情報フォーマット

- ・ 各日の開始時に提供されるマップ情報は次のような JSON 形式データで提供します。
- ・ "endsAt" キーに当該日の回答受付終了時刻(UNIX TIME)を記録します。
- ・ "day" キーに、マップ情報提供前の日付を記録します。初日の場合は 0 となります。
- ・ "agents"キーに、自チームのエージェント情報を記録します。"kind"キーがエージェントの種類 (0: 巡回車, 1: 補給車), "pos"キーがエージェントの位置, "fuel"キーが燃料積載量を表します。
- ・ "others"キーに、他チームのエージェント情報を記録します。"id"キーが他チームの ID を表し, "agents"キーは自チームの情報と同じ形式で情報を記録します。
- ・ "traffics"キーに、道路セルの渋滞情報を記録します。"pos"キーが道路の位置, "status"キーが各道路セルの状態(0: 順調, 1: 混雑, 2: 渋滞)を表します。

```
{
  "endsAt": 1778227205,
  "day": 1,
  "agents": [
    {"kind": 0, "pos": 1, "fuel": 20},
    {"kind": 1, "pos": 1, "fuel": 20},
    {"kind": 0, "pos": 9, "fuel": 10},
    {"kind": 0, "pos": 9, "fuel": 0}
  ],
  "others": [
    {
      "id": 0,
      "agents": [
        {"kind": 0, "pos": 1, "fuel": 2},
        {"kind": 0, "pos": 1, "fuel": 3},
        {"kind": 0, "pos": 1, "fuel": 4},
        {"kind": 0, "pos": 1, "fuel": 5}
      ]
    },
    {
      "id": 1,
      "agents": [
        {"kind": 1, "pos": 1, "fuel": 20},
        {"kind": 1, "pos": 9, "fuel": 20},
        {"kind": 1, "pos": 17, "fuel": 20},
        {"kind": 1, "pos": 25, "fuel": 20}
      ]
    }
  ],
  "traffics": [
    {"pos": 1, "status": 0},
    {"pos": 9, "status": 0},
    {"pos": 17, "status": 1},
    {"pos": 25, "status": 2}
  ]
}
```

## エージェント種別の回答フォーマット

- ・ 試合開始前に、エージェントの種類を次のような JSON 形式データで受け取ります。
- ・ エージェントの種類(0:巡回車, 1:補給車)を配列下に記述します。配列の要素数はエージェント数と一致する必要があります。
- ・ 要素数がエージェント数と異なっていたり、エージェントの種類に {0, 1} 以外が指定されているデータは不正データとし、リジェクトします。制限時間内に正しいデータが受理されなかった場合は、すべてのエージェントを巡回車とします。

```
[0, 1, 0, 1]
```

## 行動計画の回答フォーマット

- ・ 回答では各日の行動計画を次のような JSON 形式データで受け取ります。
- ・ 全エージェント分の行動データを配列下に記述します。例では 2 エージェントの行動列を記述しています。
- ・ -1 以下の値は待機を表します。例の[-15] では 15 ステップ待機を指定しています。
- ・ 0~5 は移動方向を表します。0 が左上方向への移動, 1 が右上方向への移動…(以降は時計回り)となります。
- ・ 例の[0,1,-10]は左上に移動した後に右上に移動し、その後 10 ステップ待機を指定しています。
- ・ 各エージェントの行動計画は 1 日のステップ数と一致する必要があります。また、移動不可能なセル(池・マップ外)への移動が含まれる回答は不正データとします。1 エージェントでも行動計画が不正な場合は、すべてのエージェントの行動計画を不正とし、リジェクトします。制限時間内に正しいデータが受理されなかった場合は、すべてのエージェントが 1 日待機とします。

```
[  
  [-15],  
  [0,1,-10]  
]
```