

プログラミング演習 最終課題レポート

08-252019 小倉直己

1 選択した課題

今回は、潜水艦ゲームの課題を選択した。

2 AI のアルゴリズムの説明

今回作成した AI は、以下のアルゴリズムに則って動作するようにした。

2.1 船の配置

味方の艦の配置は、全ての艦が以下のルールを満たすように配置される。

- 艦は 5×5 のマス目の中に配置される。
- ある艦から半径 2 マスの正方形の範囲内に他の艦が存在しない。

この配置を実装すると、以下のプログラムのようになる。

```
def place_ship(self):
    distance = 2 # 2 マス以上離す
    placed_positions = set()
    ship_types = ['w', 'c', 's']
    max_attempts = 500
    for ship_type in ship_types:
        placed = False
        attempts = 0
        while not placed and attempts < max_attempts:
            attempts += 1

            # choice a random position which is not occupied, and scattered.
            row = self.rng.randint(0, self.field.height - 1)
            col = self.rng.randint(0, self.field.width - 1)
            position = [col, row]

            if tuple(position) not in placed_positions:
```

```

        is_isolated = True
        for placed_pos in placed_positions:
            if abs(position[1] - placed_pos[1]) <= distance and abs(position[0] - placed_pos[0]) <= distance:
                # If the new position is too close to any placed ship, break
                is_isolated = False
                break

        if is_isolated:
            placed_positions.add(tuple(position))
            placed = True
            logging.info(f"Placed {ship_type} at {position}")
        if not placed:
            logging.warning(f"Failed to place {ship_type} after {max_attempts} attempts. Trying again")
            for r in range(self.field.height):
                for c in range(self.field.width):
                    if (c, r) not in placed_positions:
                        placed_positions.add((c, r))
                        logging.info(f"Placed {ship_type} at {[c, r]}")
                        break
            if placed:
                break

    return {
        'w': list(list(placed_positions)[0]),
        'c': list(list(placed_positions)[1]),
        's': list(list(placed_positions)[2])
    }

```

このコードは、今回作成した `original_player.py` の一部であり、`OriginalPlayer` クラスの `place_ship` メソッドとして実装されている。このプログラムでは、ランダムに艦を配置する試行を 100 回繰り返し、先ほど述べたルールに従って配置できなかった場合は、ルールと関係なくランダムに配置するようにしている。