

# 第1回 C++ プログラミング II 実験

出題日：2021 年 4 月 9 日（金）15:00 JST

第一提出期限：2021 年 4 月 9 日（金）16:40 JST

第二提出期限：2021 年 4 月 15 日（木）23:59 JST

提出方法：コースパワーの指定された箇所に提出

## 実験課題 1

コースパワーにアップロードされている 3 つのファイル (robot.hpp, prac01-robot-impl-skel.cpp, prac01-main-skel.cpp) をダウンロードして、以下の問題をよく読み、プログラムを完成させなさい。

### 問題

Robot クラス変数は、 $x$  座標と  $y$  座標およびフレーズ（セリフ）を保持する。また、Robot クラス変数には記憶しているフレーズを標準出力に書き出す機能と、他のロボットとの距離を測る機能がある。

ダウンロードした prac01-robot-impl-skel.cpp と prac01-main-skel.cpp とを編集して以下の関数を実装しなさい。

1. void Robot::set\_pos(double, double);
2. void Robot::set\_phrase(const std::string&);
3. double Robot::get\_distance\_to(const Robot&) const;
4. bool audible(double, const Robot&, const Robot&);

set\_pos 関数は、あるロボットの 2D 座標を設定することができる Robot クラスのメンバ関数である。第一引数で  $x$  座標を、第二引数で  $y$  座標をそれぞれ指定する。set\_phrase 関数は、あるロボットにフレーズを保持させるための Robot クラスのメンバ関数である。get\_distance\_to 関数は、あるロボットから引数で指定されるロボットまでの距離<sup>a</sup>を返す Robot クラスのメンバ関数である。audible 関数は 2 つのロボット間の距離が第一引数未満であれば真を、そうでなければ偽を返す関数である。

実装後にコンパイルして得られた実行ファイルから、以下の実行結果が得られることを確認しなさい。ただし下線はキーボードからの入力を示す。また、複数ファイルのコンパイル方法については講義資料を参考にする。

<sup>a</sup> このときの距離はユークリッド距離とする。すなわち、ロボット  $r_1$  の位置  $p_1 = (x_1, y_1)$  と、ロボット  $r_2$  の位置  $p_2 = (x_2, y_2)$  が与えられたとき、それらロボット間の距離  $p_1 p_2$  は  $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$  と等しい。

### 実行例 1

```
./prac01
Input taro's location: 0 0
I am taro
My UUID is 0
My UUID is 1
My UUID is 3
```

### 実行例 2

```
./prac01
Input taro's location: 0 2
I am taro
My UUID is 0
My UUID is 1
My UUID is 2
My UUID is 3
My UUID is 4
```

### 実行例 3

```
./prac01
Input taro's location: 2 2
I am taro
My UUID is 0
My UUID is 1
My UUID is 2
My UUID is 3
```

ソースコード 1 robot.hpp (変更の必要なし)

```

1 #include <string>
2
3 // 2D平面上にあるロボットであり，直交座標系で表現されるxy座標とフレーズ（セリフ）を保持する
4 // 記憶しているフレーズを標準出力に書き出したり，他のロボットとの距離を測ったりできる
5 class Robot {
6
7     private:
8         double x{}, y{};
9         std::string phrase{};
10
11     public:
12
13         // x座標とy座標を設定する
14         void set_pos(double, double);
15
16         // フレーズを記憶させる
17         void set_phrase(const std::string&);
18
19         // 記憶しているフレーズを標準出力に書き出す
20         void say() const;
21
22         // 引数に与えられたロボットとの距離を返す
23         double get_distance_to(const Robot&) const;
24 };

```

ソースコード 2 prac01-robot-impl-skel.cpp

```

1 #include <iostream>
2 #include <cmath> // あるロボットから別のロボットまでの距離を返すメンバ関数での平方根計算で
                 // std::sqrtを利用してもよい
3 #include "robot.hpp"
4
5 // ロボットのx座標とy座標を設定するためのRobotクラスのメンバ関数を実装する
6 /*
7  * 空欄1
8  */
9
10 // ロボットのフレーズ（セリフ）を設定するためのRobotクラスのメンバ関数を実装する
11 /*
12  * 空欄2
13  */
14
15 // ロボットのフレーズ（セリフ）を標準出力に書き出すためのRobotクラスのメンバ関数
16 void Robot::say() const{
17     std::cout << phrase << std::endl;
18 }
19
20 // あるロボットから別のロボットまでの距離を返すメンバ関数を実装する
21 /*
22  * 空欄3
23  */

```

```
1 #include <iostream>
2 #include <vector>
3 #include "robot.hpp"
4
5 // ロボット間の距離一定未満であればtrueを, そうでなければfalseを返す関数を作成する
6 /*
7  * 空欄4
8  */
9
10 // ロボットクラスを使うメイン関数
11 int main(){
12
13     // ロボット太郎を作成
14     Robot taro;
15
16     // 太郎の位置を決める
17     std::cout << "Input_taro's_location:_";
18     double x{}, y{};
19     std::cin >> x >> y;
20     taro.set_pos(x, y);
21
22     // 太郎に自己紹介をさせる
23     taro.set_phrase("I_am_taro");
24     taro.say();
25
26     // 太郎以外にもロボットを作成する
27     std::vector<Robot> robots(5);
28     robots[0].set_pos(0, 0);
29     robots[1].set_pos(1, 1);
30     robots[2].set_pos(2, 4);
31     robots[3].set_pos(-1, 1);
32     robots[4].set_pos(-2, 4);
33     for(auto i=0; i<robots.size(); ++i)
34         robots[i].set_phrase("My_UUID_is_" + std::to_string(i));
35
36     // 太郎と他のロボット間の距離が一定未満であれば, そのロボットのフレーズを標準出力に書き出す
37     const double audible_distance{4.0};
38     for(const auto& r: robots)
39         if(audible(audible_distance, taro, r))
40             r.say();
41
42     return 0;
43 }
```