

# CHaser2011

## ステップアップヒント4

### 15 サンプルプログラム4(ステータスを表示したプログラム)

#### ①プログラムの保存先

program フォルダの下に「sample2011\_04」という名前のフォルダを作り、次のプログラムを入力します。プログラム名は「sample2011\_04.java」です。

#### ②プログラムの動作

- ・ クール専用のプログラムで、基本の動きは sample2011\_03 と同じです。
- ・ ポート番号が 2009 のときはクール、2010 のときはホットで動作するようにモード番号を格納してからプログラムを開始します。
- ・ ステータス (ターン数、モード番号、アイテム数) を画面に表示させます。

#### ③サンプルプログラム4 (sample2011\_04.java)

```
1  /*****
2  sample2011_04.java
3  *****/
4  public class sample2011_04 {
5      public static void main(String[] args) {
6          int[] value;
7          value = new int[10];
8
9          int mode = 1, // 現在のモード
10         old_mode = 1, // 前のモード
11         turn = 0, // ターン数
12         item = 0; // アイテム数
13
14         /***** 競技サーバに接続する *****/
15         edu.procon.Connect2010 target;
16         target = new edu.procon.Connect2010("サンプ 4", args[0], Integer.parseInt(args[1]));
17
18         if(Integer.parseInt(args[1]) == 2009) {
19             mode = 1;
20             System.out.print("クールで競技サーバに接続しました。");
21         }else{
22             mode = 3;
23             System.out.print("ホットで競技サーバに接続しました。");
24         }
25
26         while (true) {
27             value = target.getReady();
28             if (value[0] == 0) break;
29             turn++; // ターン数+1
30
31             // ステータス表示
32             System.out.print("¥turn=" + turn + ", mode=" + mode + ", item=" + item);
33
34             /***** 周囲 (上下左右) に相手がいるかチェック *****/
35             if(value[2]==1 || value[4]==1 || value[6]==1 || value[8]==1){
```

```

36     mode = 90;           // mode を 90 に変更する
37 }
38 /***** 周囲（上下左右）にアイテムがあるかチェック *****/
39 if(value[2]==3 || value[4]==3 || value[6]==3 || value[8]==3){
40     old_mode = mode;    // 現在のモードを old_mode に保存する
41     mode = 20;         // mode を 20 に変更する
42 }
43 /***** mode の値で分岐する *****/
44 switch (mode) {
45     case 1:              // ブロック（壁）にぶつかるまで下に移動する
46         if(value[8] != 2){ // 下が壁でなければ、
47             value = target.walkDown(); // 下に移動する
48         }else{
49             value = target.walkRight(); // 下が壁ならば、右に移動し、
50             mode = 2;           // mode を 2 に変更する
51         }
52         break;
53     case 2: // ブロック（壁）にぶつかるまで右に移動する
54         if(value[6] != 2){
55             value = target.walkRight();
56         }else{
57             value = target.walkUp();
58             mode = 3;
59         }
60         break;
61     case 3: // ブロック（壁）にぶつかるまで上に移動する
62         if(value[2] != 2){
64             value = target.walkUp();
65         }else{
66             value = target.walkLeft();
67             mode = 4;
68         }
69         break;
70     case 4: // ブロック（壁）にぶつかるまで左に移動する
71         if(value[4] != 2){
72             value = target.walkLeft();
73         }else{
74             value = target.walkDown();
75             mode = 1;
76         }
77         break;
78     case 20: // 周囲にアイテムがあったら、取りに行く
79             // 上にアイテムがあったら、上に移動する
80             if(value[2] == 3) value = target.walkUp();
81             // 左にアイテムがあったら、左に移動する
82             else if(value[4] == 3) value = target.walkLeft();
83             // 右にアイテムがあったら、右に移動する
84             else if(value[6] == 3) value = target.walkRight();
85             // 下にアイテムがあったら、下に移動する
86             else value = target.walkDown();
87             mode = old_mode; // モードを元に戻す
88             item++; // アイテム数 + 1
89             break;
90     case 90: // 周囲に相手がいたら、ブロックをのせる
91             // 上に相手がいたら、上にブロックをのせる
92             if(value[2] == 1) value = target.putUp();
93             // 左に相手がいたら、左にブロックをのせる
94             else if(value[4] == 1) value = target.putLeft();
95             // 右に相手がいたら、右にブロックをのせる

```

```

96     else if(value[6] == 1) value = target.putRight();
97     // 下に相手がいたら、下にブロックをのせる
98     else value = target.putDown();
99     break;
100    }
101    /***** 制御情報が 0 だったら終了する *****/
102    if(value[0] == 0) break;
103    }
104    /***** 競技サーバから切断する *****/
105    target.exit();
106    }
107    }

```

### ⑤プログラム説明

#### ・変数の宣言

```

11     turn = 0;    // ターン数
12     item = 0;   // アイテム数

```

ターン数とアイテム数を格納する変数を宣言します。

#### ・チーム名

```

16     target = new edu.procon.Connect2010("サンプル4", args[0], Integer.parseInt(args[1]));

```

edu.procon.Connect2010 クラスを使って競技サーバに接続します。どのプログラムが対戦しているかわかるようにするため、チーム名を「サンプル4」にしました（変更しても構いません）。

#### ・クールとホットのどちらで動作するか

```

29     if(Integer.parseInt(args[1]) == 2009) {
30         mode = 1;
31         System.out.print("クールで競技サーバに接続しました。");
32     }else{
33         mode = 3;
34         System.out.print("ホットで競技サーバに接続しました。");
35     }

```

クライアントプログラムを起動するときに指定したポート番号は args[1] の中に格納されるので if 文を使い 2009 の時はモード 1（下に移動）、2010 の時はモード 3（上に移動）するようにします。

#### ・ステータス表示

```

29     turn++; // ターン数 + 1
30
31     // ステータス表示
32     System.out.print("¥turn=" + turn + ", mode=" + mode + ", item=" + item);

```

ターン数をプラス 1 した後、ステータスを表示します書式は変更して構いません。

・アイテムを取った時

```
88 | item++; // アイテム数+1
```

獲得したアイテム数は item に格納されているのでプラス 1 します。

#### 16 サンプルプログラム4を起動するバッチファイルの作成

「サンプルプログラム4をクールで起動するバッチファイル」を作ります。

program フォルダの下に、CHaser2011sample04Cool.bat を作り、メモ帳などで画面 29 のように入力します。同様にホットで起動する CHaser2011sample04Hot.bat を作ります。

```
@echo off
echo -----
echo sample2011_04 起動用バッチファイル (JDK6 Update24) 2011.7.26
echo -----

set JAVA_HOME=C:\Program Files\Java\jdk1.6.0_24
set PATH=%PATH%;%JAVA_HOME%\bin;
set CLASSPATH=.;%JAVA_HOME%\program\edu2010b.jar
C:
cd %JAVA_HOME%\program\sample2011_04
javac sample2011_04.java
start /b java sample2011_04 localhost 2009
```

クール(2009)で実行する

【CHaser2011sample04Cool.bat】

なお、ホットを起動させる CHaser2011sample04Hot.bat は最後の行の 2009 を 2010 へ変更 してください。また、これらのバッチファイルは 1 台のパソコンでサーバと 2 つのクライアントを起動する場合 です。サーバがネットワークで接続された別のパソコンである場合は、最後の行 localhost をサーバの IP アドレスへ変更 してください。(例：サーバの IP アドレスが例えば 192.168.0.1 の場合、localhost を 192.168.0.1 へ変更する)

バッチファイルには javac でコンパイルをしていますので、プログラムを変更したときもその都度手でコンパイルする必要はありませんので、とても便利です。

※ステップアップヒントは以上で終了です。皆さんの参加をお待ちしています！