

Assignment11_1.java

```
1  //// Assignment11_1.java
2  //
3  // 課題1 : チェッカー
4  //////////////////////////////////
5  public class Assignment11_1
6  {
7      public static void main(String[] args)
8      {
9          int i,j;
10
11         // チェッカー用の2次元配列
12         int[][] checker=new int[20][20];
13
14         // チェッカーパターンを書き込む
15         for(i=0;i<checker.length;i++)
16         {
17             for(j=0;j<checker[i].length;j++)
18             {
19                 if((i+j)%6==0 || (i-j)%6==0)checker[i][j]=1;
20                 else checker[i][j]=0;
21             }
22         }
23
24         // チェッカーパターンを表示する
25         for(i=0;i<checker.length;i++)
26         {
27             for(j=0;j<checker[i].length;j++)
28             {
29                 if(checker[i][j]==1)System.out.print("#");
30                 else System.out.print(".");
31             }
32             System.out.println();
33         }
34     }
35 }
36
```

Assignment11_2.java

```
1  //// Assignment11_2.java
2  //
3  // 課題2 : 転置行列
4  ///////////////////////////////////
5  public class Assignment11_2
6  {
7      public static void main(String[] args)
8      {
9          int i,j;
10         int[][] array = {{ 1, 2, 3, 4},
11                         { 5, 6, 7, 8},
12                         { 9,10,11,12},
13                         {13,14,15,16}};          // 4行4列行列
14         int[][] t_array = new int[4][4]; // 転置行列
15
16         // 処理内容の出力
17         System.out.println("転置行列を求めます");
18
19         // 行列の出力
20         System.out.println("行列 : ");
21         for(i=0;i<array.length;i++)
22         {
23             for(j=0;j<array[i].length;j++)
24                 System.out.printf("%2d ",array[i][j]);
25             System.out.printf("\n");
26         }
27
28         // 行列の転置
29         for(i=0;i<array.length;i++)
30             for(j=0;j<array[i].length;j++)
31                 t_array[j][i]=array[i][j];
32
33         // 転置行列の出力
34         System.out.println("転置行列 : ");
35         for(i=0;i<t_array.length;i++)
36         {
37             for(j=0;j<t_array[i].length;j++)
38                 System.out.printf("%2d ",t_array[i][j]);
39             System.out.printf("\n");
40         }
41     }
42 }
43
44
```

Assignment11_3.java

```
1  //// Assignment11_3.java
2  //
3  // 課題3：行列の積
4  ///////////////////////////////////
5  public class Assignment11_3
6  {
7      public static void main(String[] args)
8      {
9          int i,j,k;
10         int[][] array_A = {{ 1, 2, 1,-3},
11                             {-2, 1,-2, 0},
12                             { 1,-1, 1,-1}}; // 行列A
13
14         int[][] array_B = {{ 1, 1, 2},
15                             { 2,-3,-1},
16                             {-1, 2, 1},
17                             { 2, 1, 2}}; // 行列B
18
19         int[][] array_AB = new int[3][3]; // 行列の積A B
20
21         // 処理内容の出力
22         System.out.println("行列の積を求めます");
23
24         // 2つの行列を出力
25         for(i=0;i<array_A.length;i++)
26         {
27             for(j=0;j<array_A[i].length;j++)
28                 System.out.printf("%3d ",array_A[i][j]);
29             System.out.printf("\n");
30         }
31         System.out.printf("X\n");
32
33         for(i=0;i<array_B.length;i++)
34         {
35             for(j=0;j<array_B[i].length;j++)
36                 System.out.printf("%3d ",array_B[i][j]);
37             System.out.printf("\n");
38         }
39         System.out.printf("=\n");
40
41         // 積を求める配列を0で初期化
42         for(i=0;i<array_AB.length;i++)
43             for(j=0;j<array_AB[i].length;j++)
44                 array_AB[i][j]=0;
45
46         // 行列の積
47         for(i=0;i<array_A.length;i++)
48             for(j=0;j<array_B[0].length;j++)
49                 for(k=0;k<array_A[0].length;k++)
50                     array_AB[i][j]+=array_A[i][k]*array_B[k][j];
51
52         // 積の出力
53         for(i=0;i<array_AB.length;i++)
54         {
55             for(j=0;j<array_AB[i].length;j++)
56                 System.out.printf("%3d ",array_AB[i][j]);
57             System.out.printf("\n");
58         }
59     }
60 }
61
62
```

Assignment11_4.java

```
1  //// Assignment11_4.java
2  //
3  // 課題4：成績処理
4  ///////////////////////////////////
5  import java.io.*;
6
7  public class Assignment11_4
8  {
9      public static void main(String[] args) throws IOException
10     {
11         int i,j;
12         int num;                // 学生数
13         String[] student_ID;    // 各学生の学籍番号
14         int[][] score_table;    // 各学生の3科目の点数
15         int[] total;            // 各学生の点数の合計
16         int[] ranking;          // 順位
17         int rnk, id_max, max;    // 順位を求めるために使用
18
19         // キーボードの準備
20         BufferedReader br;
21         br = new BufferedReader(new InputStreamReader(System.in, "Shift-JIS"));
22
23         // 処理内容の出力
24         System.out.println("成績処理を行います");
25
26         // 学生数を入力
27         System.out.println("学生数を入力してください");
28         num=Integer.parseInt(br.readLine());
29
30         // 配列要素の確保と初期化
31         student_ID = new String[num];
32         score_table = new int[num][3];
33         total = new int[num];
34         ranking = new int[num];
35         for(i=0;i<num;i++)
36         {
37             student_ID[i]="";
38             for(j=0;j<3;j++)
39                 score_table[i][j]=0;
40             total[i]=0;
41             ranking[i]=0;
42         }
43
44         // 各学生の学籍番号と点数を入力する
45         for(i=0;i<num;i++)
46         {
47             System.out.println((i+1)+"人目：");
48             System.out.print("学籍番号?>");
49             student_ID[i]=br.readLine();
50             for(j=0;j<3;j++)
51             {
52                 System.out.print("科目" + (j+1) + "の点数?>");
53                 score_table[i][j]=Integer.parseInt(br.readLine());
54             }
55         }
```

```

56
57 // 各学生の成績を集計する
58 for(i=0;i<num;i++)
59 {
60     // 合計
61     for(j=0;j<3;j++)
62         total[i]+=score_table[i][j];
63 }
64
65 // 成績に順位をつける
66 rnk=1; // 1位から順番につける
67 while(rnk<=num)
68 {
69     max=0;
70     id_max=0;
71     for(i=num-1;i>=0;i--) // 順位がついていない学生でトップを探す
72     {
73         if(ranking[i]!=0) continue; // 順位がついている学生をスキップする
74         if(max<=total[i])
75         {
76             max=total[i];
77             id_max=i;
78         }
79     }
80     ranking[id_max]=rnk;
81     rnk++;
82 }
83
84 // 集計結果を表形式で出力する
85 System.out.println();
86 System.out.println("番号\t科目1\t科目2\t科目3\t合計\t順位");
87 for(i=0;i<num;i++)
88 {
89     System.out.print(student_ID[i]+"");
90     for(j=0;j<3;j++)
91         System.out.print(score_table[i][j]+"");
92     System.out.print(total[i]+"");
93     System.out.print(ranking[i]);
94     System.out.println();
95 }
96 }
97 }
98
99

```

Assignment11_5.java

```
1  /// Assignment11_5.java
2  //
3  // 課題5 : アスキーアート
4  ///////////////////////////////////
5  public class Assignment11_5
6  {
7      public static void main(String[] args)
8      {
9          // 数値パターン
10         int[][] figure={{0,1,0,0,0,0,0,1,0,0,0,0},
11                         {2,3,4,1,5,6,0,2,3,4,0,5,6},
12                         {7,1,8,8,6,0,0,7,1,8,8,6,0},
13                         {0,8,3,9,0,0,0,0,8,3,9,0,0}};
14
15         // 2次元行列の表示
16         for(int i=0;i<figure.length;i++)
17         {
18             for(int j=0;j<figure[i].length;j++)
19             {
20                 // 各数値に対応する文字を出力
21                 switch(figure[i][j])
22                 {
23                     case 0: System.out.print(' ');break;
24                     case 1: System.out.print('_');break;
25                     case 2: System.out.print('<');break;
26                     case 3: System.out.print('\');break;
27                     case 4: System.out.print(')');break;
28                     case 5: System.out.print(',');break;
29                     case 6: System.out.print('/');break;
30                     case 7: System.out.print('(');break;
31                     case 8: System.out.print('=');break;
32                     case 9: System.out.print('-');break;
33                 }
34             }
35
36             // 改行は一行毎に入れる
37             System.out.println();
38         }
39     }
40 }
```

Assignment11_6.java

```
1  /// Assignment11_6.java
2  //
3  // 課題6 : ドリルの採点
4  //////////////////////////////////
5  public class Assignment11_6
6  {
7      public static void main(String[] args)
8      {
9          // ドリル答案
10         int[][] ans={{2,3,5},
11                     {6,8,11},
12                     {-5,2,-3},
13                     {7,7,14},
14                     {-2,-6,8}};
15
16         // ドリルの表示と採点
17         for(int i=0;i<ans.length;i++)
18         {
19             if(ans[i][0] + ans[i][1] == ans[i][2])
20                 System.out.print("o");
21             else
22                 System.out.print("x");
23
24             System.out.println("  問" + (i+1) + "    " + ans[i][0] + "+" + ans[i][1] + "=" + ans[i][2]);
25         }
26     }
27 }
```

Assignment11_7.java

```
1  /// Assignment11_7.java
2  //
3  // 課題7 : 迷路脱出
4  //////////////////////////////////
5
6  import java.util.*;
7
8  public class Assignment11_7{
9      public static void main(String[] args){
10         int[][] map={        // マップデータ
11             {1,0,1,1,1},      // - 1は壁
12             {1,0,0,0,1},      // - 0は道
13             {1,1,1,0,1},      // - 2はゴール
14             {1,0,0,0,1},
15             {1,1,2,1,1}};
16         int[] user_pos={0,1};    // 現在位置
17         int[] user_pos_can=new int[2]; // 次の移動先候補
18         char key;                // 移動用キー入力
19         boolean ok;              // 移動先に移動可能か?
20
21         Scanner scr=new Scanner(System.in,"Shift-JIS");
22
23         do{
24             // 描画
25             for(int i=0; i<map.length; i++){
26                 for(int j=0; j<map[i].length; j++){
27                     if(user_pos[0]==i && user_pos[1]==j){
28                         System.out.print("@");
29                     }else{
30                         switch(map[i][j]){
31                             case 0: System.out.print(".");break;
32                             case 1: System.out.print("#");break;
33                             case 2: System.out.print("G");break;
34                         }
35                     }
36                 }
37                 System.out.println();
38             }
39
40             // ゴールのチェック
41             if(map[user_pos[0]][user_pos[1]]==2){
42                 System.out.println("ゴールです!!");
43                 break;
44             }
45
46             // キー入力と移動候補座標
47             user_pos_can[0]=user_pos[0];
48             user_pos_can[1]=user_pos[1];
49             key=scr.next().charAt(0);
50             switch(key){
51                 case 'w': user_pos_can[0]-=1;break;
52                 case 'a': user_pos_can[1]-=1;break;
53                 case 's': user_pos_can[0]+=1;break;
54                 case 'd': user_pos_can[1]+=1;break;
55             }
```

```
56
57 // 移動候補座標のチェック
58 ok=true;
59
60 if(!(user_pos_can[0] >=0 && user_pos_can[0] < map.length && user_pos_can[1] >=0 &&
user_pos_can[1] < map[0].length)){
61     ok=false;
62 }
63
64 if(ok==true && map[user_pos_can[0]][user_pos_can[1]]==1){
65     ok=false;
66 }
67
68 // 現在地の更新
69 if(ok==true){
70     user_pos[0]=user_pos_can[0];
71     user_pos[1]=user_pos_can[1];
72 }
73 }while(true);
74 }
75 }
76
```

Assignment11_8.java

```
1  //// Assignment11_8.java
2  //
3  // 課題8 : 宝探しゲーム
4  //////////////////////////////////
5
6  import java.io.*;
7
8  public class Assignment11_8
9  {
10     public static void main(String[] args) throws IOException
11     {
12         BufferedReader br;
13         br = new BufferedReader(new InputStreamReader(System.in, "Shift-JIS"));
14
15         int[][] map=new int[5][5]; // 宝地図
16         int trow, tcol;           // 宝の場所
17         int row, col;             // 入力する場所
18         boolean gotit=false;      // 見つかったかどうかのフラグ
19
20         // 地図に 0 を代入します
21         // 0 - 掘っていない場所
22         // 1 - 掘ったとき、何もなかった場所
23         // 2 - 掘ったとき、お宝があった場所
24         for(int i=0;i<5;i++)
25             for(int j=0;j<5;j++)
26                 map[i][j]=0;
27
28         // 宝の場所をランダムに決定します
29         trow = (int)(5*Math.random());
30         tcol = (int)(5*Math.random());
31
32         do{
33             // 地図を表示します
34             for(int i=0;i<5;i++)
35             {
36                 for(int j=0;j<5;j++)
37                 {
38                     if(map[i][j]==0)System.out.print("#");
39                     else if(map[i][j]==1)System.out.print("-");
40                     else System.out.print("$");
41                 }
42                 System.out.println();
43             }
44
45             // 宝が見つかったら終了します
46             if(gotit==true)break;
47
48             System.out.println("\n掘る場所を入力してください");
49             // 行[0~4]と列[0~4]を入力します
50             do{
51                 System.out.println("行[0~4] ?");
52                 row = Integer.parseInt(br.readLine());
53             }while(row<0 || row>4);
54             do{
55                 System.out.println("列[0~4] ?");
```

```

56         col = Integer.parseInt(br.readLine());
57     }while(col<0 || col>4);
58
59     // 地図の更新をしてメッセージを表示します
60     if(row==trow && col==tcol) // 見つかった場合
61     {
62         map[row][col]=2;
63         System.out.println("●お宝発見♪");
64         // 終了するようにフラグを設定します
65         gotit=true;
66     }
67     else // 見つからなかった場合
68     {
69         map[row][col]=1;
70         int tmpr = row-trow;
71         int tmpc = col-tcol;
72         // 探した場所の周囲（上下左右斜め）にある場合はお知らせします
73         if( tmpr>=-1 && tmpr<=1 && tmpc>=-1 && tmpc<=1)
74             System.out.println("●ピッピッ!!");
75     }
76 }while(true);
77 }
78 }
79

```

Assignment11_9.java

```
1  //// Assignment11_9.java
2  //
3  // 課題9：ドット描画
4  //////////////////////////////////
5
6  import java.io.*;
7
8  public class Assignment11_9
9  {
10     public static void main(String[] args) throws IOException
11     {
12         // キーボードの準備
13         BufferedReader br;
14         br = new BufferedReader(new InputStreamReader(System.in, "Shift-JIS"));
15
16         // ドット画面を保持する2次元配列
17         // - 0:○、1:●を表す
18         int[][] dots = new int[8][8];
19
20         // 始点座標と終点座標
21         int sx, sy, dx, dy;
22
23         // 全てゼロで初期化
24         for(int i=0;i<dots.length;i++)
25             for(int j=0;j<dots[i].length;j++)
26                 dots[i][j] = 0;
27
28         do{
29             // ドット画面の表示
30             for(int i=0;i<dots.length;i++)
31             {
32                 for(int j=0;j<dots[i].length;j++)
33                     System.out.print(dots[i][j]==0?" ":"*");
34                 System.out.println();
35             }
36
37             // 終了メニュー
38             System.out.print("線を引きますか?(y or else)");
39             char ans = br.readLine().charAt(0);
40             if(ans != 'y')break;
41
42             // 座標の入力
43             do{
44                 System.out.print("始点x座標を入力してください");
45                 sx = Integer.parseInt(br.readLine());
46             }while(sx<0 || sx>=dots[0].length);
47             do{
48                 System.out.print("始点y座標を入力してください");
49                 sy = Integer.parseInt(br.readLine());
50             }while(sy<0 || sy>=dots.length);
51             do{
52                 System.out.print("->終点x座標を入力してください");
53                 dx = Integer.parseInt(br.readLine());
54             }while(dx<0 || dx>=dots[0].length);
55             do{
```

```

56         System.out.print("->終点y座標を入力してください");
57         dy = Integer.parseInt(br.readLine());
58     }while(dy<0 || dy>=dots.length);
59
60
61     // 直線の描画
62     double vecx, vecy;
63     double len;
64     // - 始点から終点のベクトル
65     vecx = dx - sx;
66     vecy = dy - sy;
67     // - 単位ベクトルを計算
68     len = Math.sqrt(vecx*vecx+vecy*vecy);
69     vecx /= len;
70     vecy /= len;
71     // - 始点に単位ベクトルを加算しながらドットを塗りつぶし
72     double cx, cy;
73     cx = sx;
74     cy = sy;
75     while(true)
76     {
77         dots[(int)(cy+0.5)][(int)(cx+0.5)] = 1;
78         cx += vecx;
79         cy += vecy;
80         if(Math.sqrt((cx-sx)*(cx-sx)+(cy-sy)*(cy-sy))>len+0.5)break;
81     }
82 }while(true);
83 }
84 }
85

```