

```
1 // 01 8進10進変換
2 import java.io.*;
3 public class Code01{
4     public static void main(String[] args) throws IOException{
5         BufferedReader br;
6         br=new BufferedReader(new InputStreamReader(System.in));
7         System.out.println("8進数を10進数に変換します");
8         int[] octal=new int[4];
9         System.out.println("4桁の8進数を上位より入力してください");
10        for(int i=octal.length-1; i>=0; i--){
11            System.out.print((i+1)+"桁目の値は?>");
12            octal[i] = Integer.parseInt(br.readLine());
13        }
14        int value10 = octal[octal.length-1];
15        for(int i=octal.length-2; i>=0; i--){
16            value10 = value10*8 + octal[i];
17        }
18        System.out.print("8進数");
19        for(int i=octal.length-1; i>=0; i--){
20            System.out.print(octal[i]);
21        }
22        System.out.println("は10進数で"+value10+"です");
23    }
24 }
```

```

1 // 02 計算ドリル
2 import java.io.*;
3 public class Code02{
4     public static void main(String[] args) throws IOException{
5         BufferedReader br;
6         br=new BufferedReader(new InputStreamReader(System.in));
7         System.out.println("計算問題を出題します");
8         int counter = 0;
9         int miscount = 0;
10        boolean flag_exit = true;
11        char[] ops={'+', '-', '*', '/'};
12        while(flag_exit){
13            counter++;
14            System.out.println(counter+" 問目");
15            int a = 1 + (int) (Math.random()*9);
16            int b = 1 + (int) (Math.random()*9);
17            int op = (int) (Math.random()*4);
18            int ans = 0;
19            switch(op){
20                case 0: ans = a + b; break;
21                case 1: ans = a - b; break;
22                case 2: ans = a * b; break;
23                case 3: ans = a / b; break;
24            }
25            int c;
26            miscount = -1;
27            do{
28                miscount++;
29                if(miscount == 2) {
30                    flag_exit = false;
31                    break;
32                }
33                System.out.print(" "+a+ops[op]+b+" = ? ");
34                c = Integer.parseInt(br.readLine());
35            }while(ans != c);
36        }
37        System.out.println("終了します");
38    }
39 }

```

```
1 // 03 面積計算
2 import java.util.*;
3 public class Code03{
4     public static void main(String[] args){
5         System.out.println("正弦の面積を求めます");
6         int dotsout = 0;
7         int dotsin = 0;
8         double x;
9         double y;
10
11         for(int i=0; i<1000000; i++)
12         {
13             x = new Random().nextDouble()*Math.PI;
14             y = new Random().nextDouble();
15
16             if(y <= Math.sin(x))dotsin++;
17             else dotsout++;
18         }
19         System.out.println("面積はおおよそ"+(Math.PI*dotsin/(dotsin+dotsout))+ "です");
20     }
21 }
```

```

1 // 04 交通シミュレーション
2 import java.io.*;
3 public class Code04{
4     public static void main(String[] args) throws IOException{
5         BufferedReader br;
6         br=new BufferedReader(new InputStreamReader(System.in));
7         System.out.println("交通シミュレーションを行います");
8         System.out.print("車の出現確率 (0~100) を入力してください\n>");
9         int rate = Integer.parseInt(br.readLine());
10        int counter = 0;
11        int[] road = new int[64];
12        int[] road_update = new int[64];
13        boolean trafficlight = false;
14        int trafficlight_pos = road.length/2;
15
16        for(int i=0; i<road.length; i++)
17            road[i] = (int) (Math.random()*100) < rate ? 1 : 0;
18
19        while(true){
20            for(int i=0; i<road.length; i++)
21                System.out.print(road[i]==0?'_':'O');
22            System.out.print('\n');
23
24            int state = 0;
25            for(int i=0; i<road.length; i++){
26                if(i == 0){
27                    state = 4*road[road.length - 1] + 2*road[i] + road[i+1];
28                }else if(i == road.length-1){
29                    state = 4*road[i-1] + 2*road[i] + road[0];
30                }else{
31                    state = 4*road[i-1] + 2*road[i] + road[i+1];
32                }
33
34                if(i != trafficlight_pos && i != trafficlight_pos+1){
35                    switch(state){
36                        case 0: case 1: case 2: case 6:
37                            road_update[i] = 0;
38                            break;
39                        case 3: case 4: case 5: case 7:
40                            road_update[i] = 1;
41                            break;
42                    }
43                }else if(i == trafficlight_pos){
44                    switch(state){
45                        case 0: case 1:
46                            road_update[i] = 0;
47                            break;
48                        case 2: case 6:
49                            road_update[i] = (trafficlight ? 1 : 0);
50                            break;
51                        case 3: case 4: case 5: case 7:
52                            road_update[i] = 1;
53                            break;
54                    }
55                }else{
56                    switch(state){
57                        case 0: case 1: case 2: case 6:
58                            road_update[i] = 0;

```

```
59         break;
60     case 4: case 5:
61         road_update[i] = (trafficlight ? 0 : 1);
62         break;
63     case 3: case 7:
64         road_update[i] = 1;
65         break;
66     }
67 }
68 }
69
70 for(int i=0; i<road.length; i++)
71     road[i] = road_update[i];
72
73 counter++;
74 if(counter % 16 == 0)
75     trafficlight = !trafficlight;
76
77 try{
78     Thread.sleep(200);
79 } catch (Exception e) {}
80 }
81 }
82 }
```

```

1 // 05 海戦ゲーム
2 import java.io.*;
3 public class Code05{
4     public static void main(String[] args) throws IOException{
5         BufferedReader br;
6         br=new BufferedReader(new InputStreamReader(System.in));
7         System.out.println("海戦ゲームを始めます");
8         int mapsize = 5;
9         int[][] map = new int[mapsize][mapsize];
10        int[][] shipmap = new int[mapsize][mapsize];
11
12        int shiplen = 3;
13        int[][] shippos = new int[2][2];
14        shippos[0][0] = (int) (Math.random()*mapsize);
15        shippos[1][0] = (int) (Math.random()*mapsize);
16        int[][] shiptail_can = new int[2][4];
17        int[][] shiptail_dir = {{0,shiplen-1,0,-(shiplen-1)}, {-(shiplen-1),0,shiplen-
18        1,0}};
19        int index = -1;
20        for(int i=0; i<4; i++){
21            int tmpx = shippos[0][0] + shiptail_dir[0][i];
22            int tmpy = shippos[1][0] + shiptail_dir[1][i];
23            if(tmpx >=0 && tmpx <= shipmap.length-1 && tmpy >=0 && tmpy <=
24            shipmap.length-1){
25                index++;
26                shiptail_can[0][index] = tmpx;
27                shiptail_can[1][index] = tmpy;
28            }
29        }
30        int tmp = (int) (Math.random()*index);
31        shippos[0][1] = shiptail_can[0][tmp];
32        shippos[1][1] = shiptail_can[1][tmp];
33
34        int unitx = (shippos[0][1] - shippos[0][0])/(shiplen-1);
35        int unity = (shippos[1][1] - shippos[1][0])/(shiplen-1);
36        for(int i=0; i<shiplen; i++){
37            shipmap[shippos[1][0] + unity*i][shippos[0][0] + unitx*i] = 1;
38        }
39
40        System.out.println("長さ"+shiplen+"の潜水艦1台の位置を探し当てましょう");
41        for(int i=0;i<map.length;i++){
42            for(int j=0;j<map[i].length;j++){
43                switch(map[i][j]){
44                    case 0: System.out.print('?');break;
45                    case 1: System.out.print('*');break;
46                    case 2: System.out.print('.');break;
47                }
48            }
49            System.out.println();
50        }
51
52        int mx,my;
53        do{
54            System.out.print("x座標の値を入力してください>");
55            mx = Integer.parseInt(br.readLine());
56            System.out.print("y座標の値を入力してください>");
57            my = Integer.parseInt(br.readLine());
58
59            if(map[my][mx] == 0){

```

```
57     if(shipmap[my][mx] == 1){
58         System.out.println("どっか〜ん、命中!");
59         map[my][mx] = 1;
60     }else{
61         boolean detected = false;
62         for(int i=-1; !detected && i<=1; i++)
63             for(int j=-1; !detected && j<=1; j++)
64             {
65                 if(my+i<0 || my+i>map.length-1 || mx+j<0 || mx+j>map.length-1)
66                     continue;
67                 if(shipmap[my+i][mx+j] == 1){
68                     detected = true;
69                 }
70             }
71         if(detected) System.out.println("ぴっぴつ、近いようだ");
72         else System.out.println("近くにはないようだ");
73         map[my][mx] = 2;
74     }
75 }else{
76     System.out.println("どうやら調査済みのようだ");
77 }
78
79 int cnt = 0;
80 for(int i=0; i<map.length; i++){
81     for(int j=0; j<map[i].length; j++){
82         switch(map[i][j]){
83             case 0: System.out.print('?'); break;
84             case 1: System.out.print('*'); cnt++; break;
85             case 2: System.out.print('.'); break;
86         }
87     }
88     System.out.println();
89 }
90 if(cnt == shiplen) break;
91 }while(true);
92 System.out.println("発見できました");
93 }
94 }
```