

Dadi.java

```
1/* TemaEsame_Schema_Inventato2: Esercizio 6 */
2
3public class Dadi {
4    /*@ spec_public @*/ int giocatore; // 0=USER, 1=PC
5    /*@ spec_public @*/ int vincitorePartita; // 0=USER, 1=PC
6    /*@ spec_public @*/ int risultatoGiocata; // 0=WINUSER, 1=WINPC, 2=PATTA
7    /*@ spec_public @*/ int statoPartita; //0=FINE, 1=INCORSO
8    /*@ spec_public @*/ int[] soldi;
9
10
11    // INVARIANTI:
12    // 1) giocatore può solo assumere i valori 0,1
13    //@ public invariant (giocatore == 0 || giocatore ==1);
14    // 2) risultato può solo assumere valori 0,1,2,-1
15    //@ public invariant (risultatoGiocata == 0 || risultatoGiocata == 1 || risultatoGiocata
16    ==2 || risultatoGiocata == -1);
17    // 3) statoPartita può solo assumere valore 0,1,-1
18    //@ public invariant (statoPartita == 0 || statoPartita ==1 || statoPartita ==-1);
19    // 4) i soldi presenti nel sistema sono sempre in totale 20
20    //@ public invariant (\sum int i; 0<=i && i<soldi.length; soldi[i]) == 20;
21    // 5) la dimensione dell'array soldi è 2
22    //@ public invariant soldi.length == 2;
23    // 6) ciascun giocatore potrà sempre avere al massimo 20€
24    //@ public invariant soldi[0]<=20 && soldi[1]<=20;
25    // 7) l'array soldi non sarà mai nullo
26    //@ public invariant soldi!=null;
27
28    // POSTCONDIZIONI:
29    // 1) entrambi i giocatori devono avere 10€ all'inizio
30    //@ ensures (\forall int i; 0<=i && i<soldi.length; soldi[i]==10);
31    // 2) la partita deve essere INCORSO all'inizio
32    //@ ensures statoPartita == 1;
33    // 3) il vincitore della partita non è ancora definito (-1)
34    //@ ensures vincitorePartita == -1;
35    // 4) nessun risultatoGiocata è ancora stato definito (-1)
36    //@ ensures risultatoGiocata == -1;
37    public Dadi() {
38        soldi = new int[]{10,10};
39        statoPartita = 1;
40        vincitorePartita=-1;
41        risultatoGiocata=-1;
42    }
43
44    // PRECONDIZIONI:
45    //@ requires 1<=dadoUtente && dadoUtente<=6 && 1<=dadoPC && dadoPC<=6;
46
47    // POSTCONDIZIONI:
48    // 1) Se vince l'utente restituisco 0
49    //@ ensures (dadoUtente > dadoPC ==> \result == 0);
50    // 2) Se vince il pc restituisco 1
51    //@ ensures (dadoPC > dadoUtente ==> \result == 1);
52
53    // 3) Se è patta restituisco 2
54    //@ ensures (dadoPC == dadoUtente ==> \result == 2);
55    public int esitoGiocata(int dadoUtente, int dadoPC) {
56        // 0=WINUSER, 1=WINPC, 2=PATTA
```

Dadi.java

```

57         if(dadoUtente > dadoPC)
58             return 0;
59         else if (dadoPC > dadoUtente)
60             return 1;
61         else
62             return 2;
63     }
64
65
66     // PRECONDIZIONI:
67     //@ requires 1<=dadoUtente && dadoUtente<=6 && 1<=dadoPC && dadoPC<=6;
68
69     // POSTCONDIZIONI:
70     // ATTENZIONE: per le postcondizioni 1),2),3) devo mettere \old(..)
71     // perché la prima volta che si verifica soldi[0]==0 oppure soldi[1]==0
72     // non setto subito le variabili "statoPartita" e "vincitorePartita" al
73     // valore che devono assumere. Questo lo faccio alla successiva chiamata
74     // del metodo "giocaTurno".
75     // 1) se uno dei due giocatori ha terminato i soldi, la partita finisce
76     //@ ensures ( \old(soldi[0]) == 0 || \old(soldi[1])==0 ==> statoPartita==0 );
77     // 2) se il vincitore è PC, allora vincitorePartita assume valore 1
78     //@ ensures ( \old(soldi[0]==0) ==> vincitorePartita==1);
79     // 3) se il vincitore è USER, allora vincitorePartita assume valore 0
80     //@ ensures ( \old(soldi[1])==0 ==> vincitorePartita==0);
81     // NB: i controlli su "risultatoGiocata" non li metto in quanto li ho
82     // già messi nel metodo "esitoGiocata(..)"
83     // 4) se risultatoGiocata=0 (WINUSER) e il gioco non è finito
84     // allora i soldi dell'utente aumentano di 1 e i soldi del PC diminuiscono di 1
85     //@ ensures ( (risultatoGiocata==0 && statoPartita==1) ==> ( soldi[0]==\old(soldi[0])+1 &&
    soldi[1]==\old(soldi[1])-1 ) );
86     // 5) se risultatoGiocata=1 (WINPC) e il gioco non è finito
87     // allora i soldi dell'utente diminuiscono di 1 e i soldi del PC aumentano di 1
88     //@ ensures ( (risultatoGiocata==1 && statoPartita==1)==> ( soldi[0]==\old(soldi[0])-1 &&
    soldi[1]==\old(soldi[1])+1 ) );
89     // 6) se la partita è PATTA (2) e il gioco non è finito
90     // i soldi di entrambi i giocatori rimangono invariati
91     //@ ensures ( (risultatoGiocata==2 && statoPartita==1) ==> ( soldi[0]==\old(soldi[0]) &&
    soldi[1]==\old(soldi[1]) ) );
92     public void giocaTurno(int dadoUtente, int dadoPC) {
93         // quando uno dei due giocatori termina i soldi,
94         // la partita termina (0=USER, 1=PC)
95         if(soldi[0] == 0 || soldi[1]==0) {
96             statoPartita=0;
97             // vince il PC (1)
98             if(soldi[0]==0)
99                 vincitorePartita = 1;
100             // vince lo user (0)
101             else
102                 vincitorePartita = 0;
103         }
104         else {
105             risultatoGiocata=esitoGiocata(dadoUtente, dadoPC);
106             // risultatoGiocata: 0=WINUSER, 1=WINPC, 2=PATTA
107             // WINUSER
108             if(risultatoGiocata==0) {
109                 // 0=USER, 1=PC
110                 soldi[0]++;

```

Dadi.java

```
111         soldi[1]-=1;
112     }
113     // WINPC
114     else if (risultatoGiocata==1) {
115         soldi[0]-=1;
116         soldi[1]+=1;
117     }
118     // PATTA = nessuno perde o guadagna soldi
119 }
120 }
121
122
123 //
124 // public static void main(String[] args) {
125 //     Dadi d = new Dadi();
126 //     System.out.println(d.soldi[0] + " " + d.soldi[1] + " statoPartita:" + d.statoPartita +
127 //         d.vincitorePartita + d.risultatoGiocata + d.soldi.length);
128 //
129 //     d.giocaTurno(6, 1);
130 //     System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
131 //         d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
132 //     System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " + d.soldi[1]);
133 //     System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
134 //     d.giocaTurno(6, 1);
135 //     System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
136 //         d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
137 //     System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " + d.soldi[1]);
138 //     System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
139 //     d.giocaTurno(6, 1);
140 //     System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
141 //         d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
142 //     System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " + d.soldi[1]);
143 //     System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
144 //     d.giocaTurno(6, 1);
145 //     System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
146 //         d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
147 //     System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " + d.soldi[1]);
148 //     System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
149 //     d.giocaTurno(6, 1);
150 //     System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
151 //         d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
152 //     System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " + d.soldi[1]);
153 //     System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
154 //     d.giocaTurno(6, 1);
155 //     System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
156 //         d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
157 //     System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " + d.soldi[1]);
158 // }
```

Dadi.java

```

159//      System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
160//      d.giocaTurno(6, 1);
161//      System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
      d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
162//      System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " +d.soldi[1]);
163//      System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
164//      d.giocaTurno(6, 1);
165//      System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
      d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
166//      System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " +d.soldi[1]);
167//      System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
168//      d.giocaTurno(6, 1);
169//      System.out.println("Vincitore: " + (d.risultatoGiocata==0 ? "WINUSER" :
      d.risultatoGiocata==1 ? "WINPC" : "PATTA") );
170//      System.out.println("Soldi USER: " + d.soldi[0] + " --- Soldi PC: " +d.soldi[1]);
171//      System.out.println("Partita finita? " + (d.statoPartita==0 ? "FINE" : "INCORSO"));
172//
173//  }
174
175
176 }
177

```

Proofs

Max. Rule Application	Method Treatment	Dependency Contract	Query Treatment	Arithmetic Treatment	Stop at
<input type="text" value="10000"/>	<input type="radio"/> Contract <input checked="" type="radio"/> Expar	<input checked="" type="radio"/> On <input type="radio"/> Off	<input checked="" type="radio"/> On <input type="radio"/> Off	<input type="radio"/> Base <input checked="" type="radio"/> DefOps	<input type="radio"/> Default <input checked="" type="radio"/> Unclos

Type	Target	Contract	Proof Reuse	Proof Result	N...	Br...	Ti...	€	€
Dadi	Dadi()	JML operation contract 0	New Proof	Closed	628	10	19...		
Dadi	giocaTurno(int...	JML operation contract 0	New Proof	Closed	22...	54	39...		
Dadi	esitoGiocata(i...	JML operation contract 0	New Proof	Closed	524	7	577		