ROADBLOCK 2

In questo secondo controllo, probabilmente avrete la vita più facile. Diamo per scontato che abbiate applicato opportunamente i principi precendenti...anche perché in caso contrario arrivare qui potrebbe essere stato difficile. Quindi i punti seguenti dovrebbero essere risolti o quasi risolti.

* **Intimità inappropriata**: (dati indipendenti).
* **Complessità ciclomatica:** Troppe diramazioni
* **Metodo troppo lungo**: Tutti i metodi hanno meno di 10 istruzioni.
* **Codice duplicato:** assente
* **Commenti inadeguati**: assenti
* **Nomi oscuri**: assenti

Ora potete controllare se avete delle caratteristiche più legate alla natura OO del programma. Potreste aver pensato in modo procedurale, cosa che vi impedirebbe di fare progressi. Ora potete occuparvi di un refactoring un po’ più profondo.

* **Classe Dio**: avete una unica classe si occupa di tutto, tipicamente all’interno del main(). SOLUZIONE: dividere le responsabilità in classi più piccole e utilizzarle nella classe interessata tramite dipendenza/aggregazione/composizione. Possibili classi: Punto, Game, Set, Giocatore.
* **Classe Pigra**: una classe che fa poco o nulla (es: una classe Main?)
* **Classe Dati**: una classe che contiene solo dati

IMPORTANTE: Tutti i test finora effettuati devono restare VERDI dopo ogni refactoring.