Soluzioni master parte teorica

Domanda 1:

- 1 punto: commenti normali con // oppure /*..*/
- 1 punto: commenti Javadoc con /** .. */
- 1 punto: i commenti normali “spariscono” dopo la compilazione
- 1 punto: i commenti Javadoc sono parte integrante delle documentazione del codice
- 1 punto: i commenti Javadoc sono strutturati e possono contenere diversi tag

Domanda 2:

- 1 punto: override quando un metodo di una sottoclasse ridefinisce un metodo ereditato
- 1 punto: l’override si ottiene implementando un metodo con la stessa firma di un metodo erreditato
- 1 punto: in questo contesto, final puo’ essere applicato a classi e metodi (sui campi non ha a che vedere con l’ereditarieta’)
- 1 punto: final su una classe semplicemente impedisce che ci sia una sottoclasse
- 1 punto: final su un metodo impedisce l’override di un metodo

Domanda 3:

- 1 punto: tramite un oggetto Class possiamo accedere a tutti metodi della classe
- 1 punto: tramite un oggetto Class possiamo chiedere un metodo particolare specificandone la firma
- 1 punto: tramite il metodo invoke possiamo invocare un metodo
- 1 punto: tramite l’invoke dobbiamo passare tutti i parametri del metodo invocato
- 1 punto: se non abbiamo accesso al metodo o il metodo e’ astratto/non esiste, otteniamo delle eccezioni ad hoc

Soluzioni master parte pratica

Esercizio 1:

- 1 punto: la classe Match ha solo due campi stringa che memorizzano i nomi delle due squadre
- 1 punto: la classe Match ha un costruttore che inizializza in maniera appropriata i nomi delle due squadre
- 1 punto: presenza dei getter per i metodi delle squadre, o i campi sono accessibili e final
- 1 punto: presenza di un metodo winner senza parametri che ritorna una stringa
- 1 punto: il metodo winner e' astratto

Esercizio 2:

- 1 punto: la classe SoccerMatch estende Match
- 1 punto: la classe SoccerMatch contiene due campi interi che memorizzano il numero di goal
- 1 punto: il numero di goal non e' esternamente accessibile
- 1 punto: il costruttore di SoccerMatch invoca il supercostruttore di Match
- 1 punto: il metodo winner e' implementato in maniera corretta

Esercizio 3:

- 1 punto: la classe SoccerTournament memorizza squadre e match in campi privati e il costruttore inizializza tali campi correttamente
- 1 punto: c'e' la definizione di una classe che definisce l'eccezione ad hoc e il costruttore dichiara che lancia tale eccezione
- 1 punto: il costruttore controlla tramite instanceof che tutte le istanze delle classe siano SoccerMatch
- 1 punto: il costruttore controlla che entrambe le squadre siano all'interno della collection che contiene i nomi delle squadre
- 1 punto: il calcolo dei punti e' corretto