Programmazione Avanzata

4 Febbraio 2020

Rispondere alle domande nello spazio disponibile. Se questo non fosse sufficiente continuare nel retro della pagina.

Nome e Cognome: _____

Matricola: _____

Parola Chiave: _____

1 1. Consideriamo la seguente dichiarazione F#:

```
let rec aFunction t f g x =
        if (t x) then aFunction t f g (g (x-1))
        else y  aFunction t f g (f (x+1))
```

Quale è il tipo della funzione aFunction?

2 2. Consideriamo la dichiarazione delle seguenti classi Java:

```
class ClasseA {

        public void m1( ) {
                System.out.println("ClasseA->m1()");
                m2();
        }

        public void m2( ) {
                System.out.println("ClasseA->m2()");
        }

}

class ClasseB extends ClasseA {

        public void m2( ) {
                System.out.println("ClasseB->m2()");
        }

}
```

Consideriamo, inoltre, la seguente porzione di codice:

```
ClasseA c = new ClasseB();
c.m1();
```

Quale è il risultato della sua esecuzione?

- ◯ Viene stampato a video:

  ```
  ClasseA->m1()
  ClasseB->m2()
  ```

- ◯ Viene stampato a video:

  ```
  ClasseA->m1()
  ClasseA->m1()
  ```

- ◯ Il codice non viene eseguito a seguito di un errore di tipo.

Motivare la risposta:

2 3. Date le definizioni di ClasseA e ClasseB dell'esercizio precedente. Consideriamo ora le seguenti definizioni:

```
class ClasseC {
        public void m( ClasseA a ) {
                System.out.println("ClasseC->m()");
        }
}

class ClasseD extends ClasseC {
        public void m( ClasseB a ) {
                System.out.println("ClasseD->m()");
        }
}
```

Consideriamo inoltre la seguente porzione di codice:

```
ClasseA c = new ClasseB();
ClasseC d = new ClasseD();
d.m(c);
```

Quale è il risultato della sua esecuzione?

- ◯ Viene stampato a video:

  ClasseC->m()

- ◯ Viene stampato a video:

  ClasseD->m()

- ◯ Il codice non viene eseguito a seguito di un errore di tipo.

Eseguendo, invece, la seguente porzione di codice:

```
ClasseB c = new ClasseB();
ClasseD d = new ClasseD();
d.m(c);
```

Quale risultato si otterrà?

- ◯ Viene stampato a video:

  ClasseC->m()

- ◯ Viene stampato a video:

  ClasseD->m()

- ◯ Il codice non viene eseguito a seguito di un errore di tipo.

Motivare la risposta:

2 4. Descrivere il principio di sostituzione di Liskov fornendo un esempio di violazione dello stesso.

2 5. In Java i *generics* consentono di parametrizzare una classe rispetto ad un dato tipo T.

(a) Che differenza c'è tra un parametro di tipo `List<Object>` ed uno di tipo `List<?>`?

(b) Quale garanzie abbiamo su una variabile dichiarata `List<? extends ClassA>`? (Dove ClassA è una classe definita nell nostra applicazione)

(c) Come possiamo usare, invece, un oggetto del tipo Function<? super ClassA,Integer>?

2 6. Consideriamo la seguente classe Java che implementa le primitive *test-and-set* di una variabile condivisa:

```
public class SharedVariable<T> {

        private T value;

        public SharedVariable( T init ) {
                this.value = init;
        }

        public synchronized void testAndSet(
                Predicate<? super T> p ,
                T value ) throws InterruptedException {
                while (!p.test(this.value)) {
                        this.wait();
                }
                this.value = value;
                this.notifyAll(); //<--- (1)
        }

}
```

Sarebbe possibile sostituire l'istruzione (1) this.notifyAll(); con this.notify();? Potrebbero insorgere problemi?