

---

**FONDAMENTI DI PROGRAMMAZIONE B**

---

*Tempo a disposizione: 30 minuti*

Nome ..... Cognome ..... Matricola .....

*Per accedere alla prova di programmazione è necessario rispondere correttamente ad almeno il 70% delle domande*

1. [C++] Se una classe non definisce esplicitamente il distruttore

- ☐ *a* viene generato un errore a tempo di compilazione
- ☐ *b* viene generato un errore a tempo di esecuzione
- ☐ *c* la classe ha un distruttore di default
- ☐ *d* non è possibile istanziare oggetti di quella classe
- ☐ *e* nessuna delle precedenti

2. [C++] Si considerino le classi A, B, C e D. La classe B è derivata da A, mentre le classi C e D sono derivate da B. Il seguente metodo `foo` della classe B

```
void shift(B obj) {...}
```

può accettare come argomenti oggetti

- ☐ *a* di tipo A, B e Object
- ☐ *b* di tipo B ma non di tipo C e D
- ☐ *c* di tipo A e B
- ☐ *d* di tipo B, C e D

3. [C++] Se una classe definisce esplicitamente il costruttore di copia, allora è opportuno

- ☐ *a* definire esplicitamente anche l'operatore di assegnamento ma non il distruttore
- ☐ *b* definire esplicitamente anche il distruttore ma non l'operatore di assegnamento
- ☐ *c* definire anche il costruttore senza parametri
- ☐ *d* definire esplicitamente anche il distruttore e l'operatore di assegnamento
- ☐ *e* nessuna delle precedenti

4. [C++] Si consideri la classe templatica `List<T>`. Allora `List<int>` è una classe derivata da `List<float>`.

☐ T ☐ F

5. [C++] Data una classe C, la dichiarazione `C c;` inizializza la variabile `c` a NULL.

☐ T ☐ F

6. [Java] Si consideri la seguente dichiarazione di attributo all'interno di una classe **A**:

```
public static int x;
```

Si indichi la risposta corretta.

- ☐ *a* è un attributo d'istanza con visibilità di package e non modificabile (dopo la sua inizializzazione nel costruttore)
- ☐ *b* è un attributo di classe con visibilità di package e non modificabile (dopo la sua inizializzazione nel costruttore)
- ☐ *c* è un attributo di classe con visibilità pubblica e modificabile
- ☐ *d* è un attributo d'istanza con visibilità pubblica e modificabile
- ☐ *e* nessuna delle precedenti

7. [Java] Si indichi la relazione corretta fra i metodi **equals** e **hashCode**

- ☐ *a* due oggetti con lo stesso **hashCode** devono essere uguali per il metodo **equals**
- ☐ *b* due oggetti che sono uguali per il metodo **equals** devono avere lo stesso **hashCode**
- ☐ *c* se si sovrascrive il metodo **equals** non è opportuno sovrascrivere anche il metodo **hashCode**
- ☐ *d* non esiste nessuna relazione fra i metodi **equals** e **hashCode**

8. [Java] Data la classe **A**, si consideri il seguente frammento di codice

```
A x = new A();  
A y = x;
```

Dopo l'esecuzione del frammento di codice riportato sopra

- ☐ *a* y fa riferimento ad un oggetto che è una copia leggera (*shallow copy*) dell'oggetto riferito da x
- ☐ *b* y fa riferimento ad un oggetto che è una copia profonda (*deep copy*) dell'oggetto riferito da x
- ☐ *c* x e y fanno riferimento allo stesso oggetto
- ☐ *d* viene sollevata un'eccezione a tempo d'esecuzione
- ☐ *e* nessuna delle precedenti

9. [Java] Un parametro di tipo di una classe generica (e.g., **Stack(T)**) può essere specializzato esclusivamente con tipi primitivi. ☐ T ☐ F

10. [Java] Data una classe **A**, l'istruzione **A a;** è equivalente all'istruzione **A a = new A();** ☐ T ☐ F