

---

**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++] 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. La seguente funzione

```
void f(B r) {...}
```

può accettare come argomenti oggetti

- ☐ a di tipo A e B
- ☐ b di tipo C e D
- ☐ c di tipo A, B e C
- ☒ d di tipo B, C e D

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

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

3. [C++] Si consideri classe templatica `Stack<T>`. La classe `Stack<float>` è una superclasse di `Stack<int>`

☐ T ☒ F

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

☐ T ☒ F

5. [C++] Ogni classe deve definire uno e un solo costruttore.

☐ T ☒ F

6. [Java] Si considerino le classi A, B e C. La classe C è derivata da B, mentre la classe B è derivata da A. La classe A definisce un metodo `foo` che la classe B ridefinisce mentre la classe C **non** ridefinisce. Si consideri il seguente frammento di codice.

```
A obj = new C();  
obj.foo();
```

- ☒ *a* viene invocato il metodo `foo` definito nella classe B
- ☐ *b* viene invocato il metodo `foo` definito nella classe A
- ☐ *c* viene sollevata una `ClassCastException`
- ☐ *d* viene rilevato un errore a tempo di compilazione
- ☐ *e* nessuna delle precedenti

7. [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

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

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

Dopo l'esecuzione del frammento di codice riportato sopra

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

9. [Java] Data una classe A, l'istruzione `A a;` è equivalente all'istruzione `A a = new A();`

☐ T ☒ F

10. [Java] La parola chiave `private` applicata ad un campo di una classe indica che il campo, una volta inizializzato nel costruttore, non può essere modificato.

☐ T ☒ F