

---

**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 un costruttore

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

2. [C++] Cosa indica l'utilizzo della parola chiave `const` in un metodo di classe (e.g., `int f() const {...}`)?

- ☐ *a* il metodo non può essere sovrascritto
- ☐ *b* il metodo è statico
- ☐ *c* il metodo non può modificare i campi della classe
- ☐ *d* nessuna delle precedenti

3. [C++] Si considerino le classi A, B e C. La classe B è derivata da A, mentre la classe C è derivata da B. La seguente funzione `fee`

```
bool fee(B obj) {...}
```

può accettare come argomenti oggetti

- ☐ *a* di tipo B e C
- ☐ *b* di tipo A, B e `Object`
- ☐ *c* di tipo A e B
- ☐ *d* esclusivamente di tipo B

4. [C++] Si supponga che la classe C contenga il metodo `int f() const {...}`. Il metodo `f` della classe C è un metodo costante. ☐ T ☐ F

5. [C++] Una classe in C++ può avere più di un distruttore. ☐ T ☐ F

6. [Java] Quando un membro di una classe è dichiarato come **protected**, allora il membro può essere accessibile

- ☐ *a* solo all'esterno della stessa package in cui è dichiarato
- ☐ *b* solo all'interno della stessa classe in cui è dichiarato
- ☐ *c* solo all'interno della stessa classe e delle sottoclassi della classe in cui è dichiarato
- ☐ *d* solo all'interno della stessa package in cui è dichiarato

7. [Java] Si considerino le classi A, B e C. Le classi C e B sono derivate da A. La classe A definisce un metodo **fee** che la classe B ridefinisce mentre la classe C **non** ridefinisce. Si consideri il seguente frammento di codice.

```
A obj = new B();  
((B) obj).fee();
```

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

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

9. [Java] L'istruzione `String str;` è equivalente all'istruzione `String str = new String();`

☐ T ☐ F

10. [Java] L'operatore `==` confronta i riferimenti quando l'operatore è applicato ad oggetti.

☐ T ☐ F