

FONDAMENTI DI PROGRAMMAZIONE B*Tempo a disposizione: 20 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 non è possibile istanziare oggetti di quella classe
- ☒ b la classe ha un costruttore di default
- ☐ c viene generato un errore a tempo di compilazione
- ☐ d viene generato un errore a tempo di esecuzione
- ☐ e nessuna delle precedenti

2. [C++] Si indichi quale dei seguenti è un esempio di costruttore di copia.

- ☐ a `A(A *other) {...}`
- ☐ b `A(A other) {...}`
- ☒ c `A(const A &other) {...}`
- ☐ d `A(A &other) const {...}`
- ☐ e 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 foo

```
bool foo(B obj) const {...}
```

può accettare come argomenti oggetti

- ☐ a di tipo A e B
- ☐ b di tipo A, B e Object
- ☒ c di tipo B e C
- ☐ d di tipo B e C solo se costanti
- ☐ e 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 di classe. ☒ T ☐ F

5. [C++] Una classe in C++ può avere al più un costruttore. ☐ T ☒ F

6. [Java] Un campo di una classe dichiarato come **final** significa

- ☒ *a* il campo può essere modificato solo una volta
- ☒ *b* il campo non può essere modificato dopo essere stato inizializzato nel costruttore
- ☐ *c* il campo deve necessariamente essere dichiarato come **static**
- ☐ *d* il campo non può essere ereditato

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();
((C) obj).fee();
```

- ☒ *a* viene sollevata una **ClassCastException** a tempo di esecuzione
- ☐ *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* y fa riferimento ad un oggetto che è una copia profonda (*deep copy*) dell'oggetto riferito da x
- ☐ *b* y fa riferimento ad un oggetto che è una copia leggera (*shallow 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] L'operatore **==** confronta i riferimenti quando l'operatore è applicato ad oggetti.

☒ T ☐ F

10. [Java] Si consideri una classe C. L'istruzione **C obj**; è equivalente all'istruzione **C obj = new C();**

☐ T ☒ F