
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

```
int move(C 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 C e delle sue sottoclassi
- ☐ e di tipo C e delle sue superclassi

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 `Set<T>`. La classe `Set<int>` è una classe derivata da `Set<float>` ☐ T ☐ F

4. [C++] Data una classe C, la dichiarazione `C c;` inizializza la variabile c all'oggetto costruito dal costruttore senza parametri della classe C ☐ T ☐ F

5. [C++] Si supponga che la classe C contenga il metodo `void foo(int x) const {...}`. Allora il metodo `foo` è costante, cioè non è permesso al metodo `foo` di cambiare gli attributi dell'oggetto su cui è invocato. ☐ T ☐ F

6. [Java] Si considerino le classi A, B e C. Le classi C e B sono derivate 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();
((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 C:

```
private int x;
```

Si indichi la risposta corretta.

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

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

☐ T ☐ F

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

☐ T ☐ F