

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 definisce esplicitamente il costruttore di copia, allora è opportuno

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

2. [C++] Si indichi quale dei seguenti è l'esempio più appropriato di operatore di assegnamento per una classe C.

- ☐ a `C& operator=(C *other) {...}`
- ☐ b `C& operator=(C other) {...}`
- ☐ c `C& operator=(C &other) const {...}`
- ☒ d `C& operator=(const C &other) {...}`

3. [C++] Si considerino le classi X, Y e Z. La classe Y è derivata da X, mentre la classe Z è derivata da Y. La seguente funzione foo

```
X foo(Y obj) const {...}
```

può accettare come argomenti oggetti

- ☐ a di tipo X e Y
- ☒ b di tipo Y e Z
- ☐ c di tipo X, Y e Object
- ☐ d di tipo Y
- ☐ e il codice non compila

4. [C++] Una classe C ha a disposizione il distruttore esclusivamente se è stato implementato almeno un costruttore.

☐ T ☒ F

5. Si consideri la classe template `stack<T>`. Allora `stack<int>` è una classe derivata da `stack<float>`.

☐ T ☒ F

6. [Java] Si considerino le classi `Forma`, `Rettangolo` e `Quadrato`. La classe `Rettangolo` estende `Forma`, mentre la classe `Quadrato` estende `Rettangolo`. Le classi di appartenenza della classe `Rettangolo` sono

- ☒ `a` `Forma` e `Object`
- ☐ `b` `Rettangolo`, `Forma` e `Object`
- ☐ `c` `Rettangolo`, `Quadrato`, `Forma` e `Object`
- ☐ `d` `Rettangolo`, `Forma`, `Quadrato`
- ☐ `e` nessuna delle precedenti

7. [Java] Si considerino le classi `A`, `B` e `C`. Le classi `C` e `B` estendono `A`. La classe `A` definisce un metodo `foo` che viene sovrascritto sia dalla classe `B` che dalla classe `C`. Si consideri il seguente frammento di codice.

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

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

8. [Java] Quale delle seguenti affermazioni è vera riguardo l'ereditarietà?

- ☐ `a` una classe può estendere più classi
- ☒ `b` una classe può implementare più interfacce
- ☐ `c` solo le classi astratte possono implementare interfacce
- ☐ `d` nessuna delle precedenti

9. [Java] Si indichi la relazione corretta fra i metodi `equals` e `hashCode`.

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

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

☐ T ☒ F