

**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 consideri il seguente programma

```
void f() {  
    throw 1;  
}  
  
int main() {  
    try {  
        f();  
    } catch(int x) {  
        cout << "1" << endl;  
    } catch(string y) {  
        cout << "2" << endl;  
    } catch(Razionale z) {  
        cout << "3" << endl;  
    }  
    return 0;  
}
```

Si indichi cosa viene stampato a video dalla funzione `main`.

☐ *a* 1 2   ☐ *b* 1   ☐ *c* 1 2 3   ☐ *d* nessuna delle precedenti

2. [C++] Si considerino le classi `Forma`, `Rettangolo` e `Quadrato`. La classe `Rettangolo` è derivata da `Forma`, mentre la classe `Quadrato` è derivata da `Rettangolo`. La seguente funzione

```
double shift(Rettangolo r) {...}
```

può accettare come argomenti oggetti

- ☐ *a* esclusivamente di tipo `Rettangolo` e delle sue superclassi  
☐ *b* di tipo `Forma`, `Rettangolo` e `Object` ma non `Quadrato`  
☐ *c* di tipo `Forma` e `Quadrato`  
☐ *d* di tipo `Rettangolo` e `Quadrato`  
☐ *e* nessuna delle precedenti

3. [C++] È possibile allocare gli oggetti esclusivamente tramite allocazione dinamica

☐ T ☐ F

4. [C++] Si consideri il template di classe `Stack<T>`. Allora `Stack<int>` è una classe derivata da `Stack<float>`

☐ T ☐ F

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

☐ T ☐ F

6. [Java] Si considerino le classi `Forma`, `Triangolo` e `Quadrato`. Le classi `Quadrato` e `Triangolo` sono derivate da `Forma`. La classe `Forma` definisce un metodo `perimetro` che le classi `Quadrato` e `Triangolo` ridefiniscono. Si consideri il seguente frammento di codice.

```
Forma f = new Triangolo();  
((Triangolo) f).perimetro();
```

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

7. [Java] Si consideri la seguente dichiarazione di attributo all'interno di una classe `C`:

```
private static final int x = 1;
```

Si indichi la risposta corretta.

- ☐ *a* è un'attributo di classe con visibilità privata e modificabile
- ☐ *b* è un'attributo d'istanza con visibilità privata e non modificabile
- ☐ *c* è un'attributo di classe con visibilità privata e non modificabile
- ☐ *d* è un'attributo di classe con visibilità di package e modificabile
- ☐ *e* nessuna delle precedenti

8. [Java] Si considerino le classi `Forma`, `Rettangolo` e `Quadrato`. La classe `Rettangolo` è derivata da `Forma`, mentre la classe `Quadrato` è derivata da `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

9. [Java] Un'eccezione di tipo controllato viene catturata e gestita automaticamente dal garbage collector.

☐ T ☐ F

10. [Java] Per rendere visibile il campo `p` di tipo `int` di una classe `C` è necessario dichiarare il campo come `package <nome_package> int p;`

☐ T ☐ F