

Esempi domande prova in itinere

1

Si descrivano i meccanismi di implementazione del polimorfismo nel C++, considerando anche il caso di costruttori e distruttori.

2

Si descrivano i livelli di accesso nelle classi C++ e come questi cambino secondo il tipo di ereditarietà.

3

Si discuta la const correctness, con particolare riferimento ai puntatori e riferimenti del C++, oltre che ai metodi costanti.

4

Cos'è una reference e quando è bene usarla? Indicare le maggiori differenze rispetto all'uso dei puntatori.

5

Date le seguenti classi e oggetti indicare qual è l'output del programma e motivare la risposta

```
#include <iostream>
class B {
public:
    virtual void foo() { std::cout << "B:foo" << std::endl; }
    void bar() { std::cout << "B:bar" << std::endl; }
};
class D : public B {
public:
    virtual void foo() override { std::cout << "D:foo" << std::endl; }
    void bar() { std::cout << "D:bar" << std::endl; }
};

int main() {
    B* pB = new B;
    pB->foo();
    pB->bar();
    delete pB;
    pB = new D;
    pB->foo();
    pB->bar();
    delete pB;
}
```

Risposta:

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
B:foo B:bar B:foo B:bar	B:foo B:bar D:foo D:bar	B:foo B:bar D:foo B:bar

6

Completare le due seguenti funzioni che sommano due argomenti, per garantire la massima performance e sicurezza.

<pre>int somma(_____ x, _____ y) { int somma = x+ y; return somma; }</pre>	<pre>template<typename __> __ somma(_____ x, _____ y) { __ somma = x + y; return somma; }</pre>
--	---