



Nome

Cognome

Matricola

Domande Traccia B

Domanda D1	<p>Si consideri il seguente metodo:</p> <pre>const int size = 4; int metodo (int m[size][size]) { int x = 0; for (int i = 0; i < size; i++) { int y = 1; for (int j = 0; j < size; j++) y *= m[i][j]; x += y; } return x / size; }</pre> <p>Quale sarà il contenuto della variabile res dopo l'esecuzione del seguente frammento di codice:</p> <pre>int m[size][size] = {{1,1,2,3}, {1,4,1,1}, {1,2,2,3}, {5,1,1,2}}; int res = metodo(m);</pre>
Risposta R1	

Domanda D2	<p>Si consideri il seguente metodo:</p> <pre>const int size = 4; int metodo (int v1[size], int v2[size]) { int x = 0; for(int i = 0; i < size; i++) if((v1[i] == v2[size-i-1])) x++; return x; }</pre> <p>Quale sarà il contenuto della variabile res dopo l'esecuzione del seguente frammento di codice:</p> <pre>int v1[size] = {1,4,2,6}; int v2[size] = {1,2,4,6}; int res = metodo(v1,v2);</pre>
Risposta R2	

Domanda D3	<p>Quali tra queste espressioni sono corrette (ed equivalenti)?</p> <ol style="list-style-type: none"> 1. <code>*(this.a);</code> 2. <code>(*this)->a;</code> 3. <code>this->a;</code> 4. <code>(*this).a;</code>
Risposta R3	

Domanda D4	<p>Si consideri il seguente metodo:</p> <pre> const int size = 4; void metodo (int v[size], int x) { for(int i = 0; i < size; i++) if(v[i] < x) v[i] = x - v[i]; else if (v[i] > x) v[i] = v[i] - x; else v[i] = x; } </pre> <p>Quale sarà il contenuto del vettore v dopo l'esecuzione del seguente frammento di codice:</p> <pre>int v[size] = {5,2,4,7}; metodo(v,4);</pre>
Risposta R4	

Data la seguente dichiarazione di classe:

```

class prova {
    private:
        int x;          // variabile intera
        string y;       // oggetto stringa STL
        char *z;        // stringa di caratteri
    public:
        prova(int a, string b, char *c);
        ~prova();
        const prova & operator= (const prova & p);
}

```

Domanda D5	Scrivere l'implementazione del distruttore di prova
Risposta R5	

Domanda D6	Scrivere l'implementazione del costruttore a 3 parametri di prova usando la lista di inizializzazione in tutti i casi in cui è possibile.
Risposta R6	

Domanda D7	Definire ed implementare l'operatore di accesso posizionale (operator[]) della classe prova in al fine di poter cambiare il contenuto della variabile z
Risposta R7	

Domanda D8	Scrivere l'implementazione dell'operatore di assegnamento di prova
Risposta R8	

Domanda D9	<p>Si considerino le seguenti dichiarazioni:</p> <pre> class A { public: void a1(); protected: void a2(); private: void a3(); }; class B: protected A { public: void b1(); protected: void b2(); private: void b3(); }; class C: private B { public: void c1() { a3(); //n.1 a2(); //n.2 a1(); //n.3 b3(); //n.4 b2(); //n.5 b1(); //n.6 } }; </pre> <p>Quali tra le istruzioni numerate sono ERRATE?</p>
Risposta R9	

Domanda D10	<p>Si considerino le seguenti istruzioni:</p> <ol style="list-style-type: none"> 1. <code>const int c = 3;</code> 2. <code>int v1[];</code> 3. <code>int *v2 = new int[3];</code> 4. <code>char *v3 = new char[3];</code> 5. <code>int v4[3] = new *int[3];</code> 6. <code>char v5[] = {'x', 'y', 'z'};</code> 7. <code>int v6[] = new int[3];</code> 8. <code>int *v7 = {1, 2, 3};</code> <p>Indicare, se presenti, la/e linea/e di codice contenente/i errore/i.</p>
Risposta R10	