



Nome

Cognome

Matricola

Domande Traccia B

Domanda D1	<p>Si consideri il seguente metodo:</p> <pre>void v2(int v[], int n) { int i = 0; while(i < n/2) { int temp = v[i]; v[i] = v[n-i-1]; v[n-i-1] = temp; i++; } }</pre> <p>Quale sarà il contenuto del vettore v dopo l'esecuzione del metodo se inizialmente vale:</p> <p>v = {2,3,4,1,5,7}</p>
Risposta R1	<ol style="list-style-type: none"> 1. v = {4,3,2,7,5,1} 2. v = {7,5,1,4,3,2} 3. v = {2,3,4,1,5,7}

Domanda D2	<p>Si consideri il seguente metodo:</p> <pre>void v3(int v[], int n) { int temp = v[0]; for(int i = 0; i < n-1; i++) v[i] = v[i+1]; v[n-1] = temp; }</pre> <p>Quale sarà il contenuto del vettore v dopo l'esecuzione del metodo se inizialmente vale:</p> <p>v = {2,3,4,1,5,7}</p>
Risposta R2	<ol style="list-style-type: none"> 1. v = {7,2,3,4,1,5} 2. v = {7,5,1,4,3,2} 3. v = {3,4,1,5,7,2}

Domanda D3	<p>Quale di queste istruzioni è ERRATA?</p> <p>(a) <code>int a1[5];</code> (b) <code>int *a2 = new int[5];</code> (c) <code>int *a4 = {1,2,3,4};</code> (d) <code>int a3[] = {1,2,3,4};</code> (e) <code>int a5[] = new int[5];</code></p>
Risposta R3	<ol style="list-style-type: none"> 1. nessuna 2. (c) and (e) 3. (d) and (e)

Domanda D4	<p>Si consideri la seguente funzione:</p> <pre>bool f2(int v[], int n) { for(int i = 0; i < n-1; i++) if(v[i] == v[i+1]) return true; return false; }</pre> <p>Cosa restituisce se il vettore v inizialmente vale:</p> <p>v = {1,2,2,3,3,3}</p>
Risposta R4	<ol style="list-style-type: none"> 1. true 2. false 3. ERRORE

Domanda D5	<p>Si consideri la seguente dichiarazione:</p> <pre>1 class Riga 2 { 3 public: 4 Riga(); 5 Riga(int size); 6 Riga(const Riga & riga); 7 ~Riga(); 8 const Riga & operator=(const Riga & riga); 9 private: 10 int *vett; 11 int size; 12 void setSize(int size); 13 };</pre> <p>Indicare la linea di codice corrispondente al prototipo del costruttore di copia</p>
Risposta R5	linea 6

Domanda D6	<p>In riferimento alla definizione di classe della domanda D5 quale sarà il codice del distruttore:</p> <p>(a) <code>Riga::~~Riga() { delete size; }</code> (b) <code>Riga::~~Riga() { delete []vett; }</code> (c) <code>Riga::~~Riga() { delete []vett; delete []size; }</code></p>
Risposta R6	<ol style="list-style-type: none"> 1. (b) or (c) 2. (a) 3. (b)

Domanda D7	<p>In riferimento alla classe della domanda D5 e dato il seguente frammento di codice:</p> <pre>Riga pippo; Riga pluto; pluto = Riga(pippo);</pre> <p>cosa viene invocato?</p>
Risposta R7	<ol style="list-style-type: none"> 1. costruttore di default → costruttore di default → operatore di assegnamento → costruttore di copia 2. costruttore ad 1 parametro → costruttore ad 1 parametro → costruttore di copia → operatore di assegnamento 3. costruttore di default → costruttore di default → costruttore di copia → operatore di assegnamento

Domanda D8	In riferimento al metodo <code>void setSize(int size);</code> della classe Riga (domanda D5 , linea 12), quale frammento di codice deve essere eseguito nel caso in cui la dimensione corrente è differente dalla nuova?
Risposta R8	1. <code>this->size = size; delete []vett; vett = new int[size];</code> 2. <code>vett = new int[size]; this->size = size; delete []vett;</code> 3. <code>delete []vett; vett = new int[this->size]; this->size = size;</code>

Domanda D9	In riferimento alla definizione di classe della domanda D5 e dato il seguente frammento di codice: <code>(a) Riga prima(3);</code> <code>(b) Riga seconda = prima;</code> <code>(c) prima = seconda;</code> <code>(d) Riga terza(prima);</code> <code>(e) terza = Riga(seconda);</code> In quali linee di codice viene invocato il costruttore di copia?
Risposta R9	1. <code>(c)</code> and <code>(e)</code> 2. <code>(b)</code> and <code>(d)</code> and <code>(e)</code> 3. <code>(b)</code> and <code>(c)</code>

Domanda D10	Si considerino le seguenti dichiarazioni: <pre> 1 class A { 2 public: 3 int pub; 4 protected: 5 int prot; 6 private: 7 int priv; 8 }; 9 class B: public A { 10 public: 11 int pubB; 12 int f() { return prot - priv; } 13 }; 14 class C: private A { 15 protected: 16 int protC; 17 int f() { return prot - priv; } 18 }; </pre> Quale di queste istruzioni è potenzialmente pericolosa?
Risposta R10	1. <code>B *puntB = new B;</code> 2. <code>A *puntA = puntB;</code> 3. <code>B *puntB_1 = static_cast<B*>(puntA);</code>