



Note allo svolgimento. Ti sono stati consegnati:

- a — il *testo del compito*;
- b — una *scheda anagrafica* con i tuoi dati: sul retro della scheda vi sono due adesivi che riportano un codice a barre;
- c — una *scheda delle risposte*;
- d — un *foglio di istruzioni* per la compilazione della scheda delle risposte.

Per un corretto svolgimento della prova, è necessario seguire in sequenza le seguenti istruzioni:

1. applicare *uno* dei due adesivi contenenti il codice a barre nell'apposito riquadro sulla scheda anagrafica;
 2. scrivere il tipo di compito assegnato (**E**) sulla scheda anagrafica, di fianco al proprio nome;
 3. compilare la scheda delle risposte, indicando sulla stessa con una croce × le risposte ritenute corrette, (vedi foglio di istruzioni). A questo proposito, è importante sapere che:
 - ogni errore determinerà un **punteggio negativo**;
 - se a un quesito non si fornisce alcuna risposta, si otterrà per quel quesito il **punteggio 0**;
 4. al termine della prova, applicare il secondo codice a barre adesivo nell'apposito riquadro sulla scheda delle risposte.
- Consegnare i due fogli su cui è stato applicato il codice a barre (scheda anagrafica e scheda delle risposte).
È severamente proibito consultare libri, appunti, manuali, o strumenti elettronici (computer, cellulari, palmari ecc.).
La prova dura 1 ora.

| | | | |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>1. [3.5]</p> | <p>Si consideri l'architettura di un Personal Computer. A Il bus consente il trasferimento di istruzioni dalla memoria alla CPU. B Il bus consente esclusivamente la trasmissione unidirezionale tra memoria e CPU. C Il clock comunica l'ora corrente ai dispositivi che la richiedono. D Durante un ciclo di esecuzione il registro PC contiene l'indirizzo dell'istruzione da eseguire nel ciclo successivo. E La memoria cache di primo livello e' volatile.</p> | <p>2. [3.5]</p> | <p>Linguaggi di programmazione. A L'esecuzione di un programma interpretato e' tipicamente piu' veloce dell'esecuzione dello stesso programma compilato. B Un programma scritto in linguaggio assembler necessita di un traduttore. C Un programma compilato e linkato e' sicuramente privo di errori. D Il debugger e' utile nella fase di sviluppo di un programma. E Le istruzioni in linguaggio assembler sono espresse mediante una sequenza di bit.</p> |
| <p>3. [4]</p> | <p>Si consideri il seguente programma C:</p> <pre>#include <stdio.h> main() { int x=2; float y=x--; char z='y'; if (--y) if(--x) z='a'; else z='b'; else z=x; /* punto 1 */ y=(x+=3, x--); /*istruzione 1*/ /* punto 2 */ { float A; A=(z>'a'? --y: y/x); /* punto 3 */ } /*punto 4*/ }</pre> <p>A Al punto 4 la variabile A non e' visibile. B Al punto 1 la variabile x ha valore 1. C L'istruzione 1 e' equivalente alla sequenza di istruzioni: x=x+3; x--; y=x; D L'istruzione 1 effettua 3 assegnamenti. E Al punto 1 la variabile z assume il valore 'b'.</p> | <p>4. [4]</p> | <p>Si consideri il seguente programma C:</p> <pre>#include <stdio.h> main() {int x; float y; char z, w; x=4; y=-2; z='B'; w='C'; x=(x%(int)y ?++x: w-z); /* istruzione 1 */ y=x++/y; /* istruzione 2 */ w=z+=1; w-=z; /* istruzione 3 */ }</pre> <p>A Immediatamente dopo l'esecuzione dell'istruzione 2, la variabile x ha il valore 2. B Immediatamente dopo l'esecuzione dell'istruzione 3, le variabili z e w hanno lo stesso valore. C Immediatamente dopo l'esecuzione dell'istruzione 3, la variabile z ha valore 'C'. D Immediatamente dopo l'esecuzione dell'istruzione 3, la variabile w ha valore '0'. E Il programma puo' essere eseguito correttamente.</p> |

| | | |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>5. [4]</p> | <p>Si consideri il seguente programma C:</p> <pre>#include <stdio.h> main() {int V[4]={0, 3, 4, 8}; int i, x, y=0, z; for(i=3; i>=0; i--)/ *istruzione 1*/ switch(V[i]%3-1) /*istruzione 2*/ {case 0: V[i]--; case 1: x=V[i]; break; case 2: z=0; break; default: y=-x; } z=V[0]+V[2] - y; /* istruzione 3*/ }</pre> <p>A Immediatamente dopo il ciclo for, la variabile z vale 0. B Immediatamente dopo l'istruzione 3, la variabile x ha il valore 1. C Al termine dell'esecuzione, il valore di uno solo tra gli elementi di V risulta modificato rispetto al suo valore iniziale. D Al termine dell'esecuzione, il valore di i e' uguale a 0. E Immediatamente dopo l'istruzione 3, la variabile z ha il valore 2.</p> | <p>Si consideri il seguente programma C:</p> <pre>#include <stdio.h> #define N 8 main() {typedef float f[N]; typedef char c[N]; f A={0,0.5,1,1.5,2.0,2.5,3.0,3.5}; c B={'C','i','a','o',0,1,2,3}; int i; A[N-1]=0; for (i=1; i<N; i+=i) A[i]=B[N-i]/i; /* punto 1 */ if (A[--i]?1:A[0]) printf("Ciao!\n"); else {i=A[0]; for (;i<5;) printf("%c", B[i++]); } /* punto 2 */ }</pre> <p>6. [4]</p> <p>A Nel programma vi sono tipi non primitivi. B Al punto 2: la variabile i ha valore 5. C Il programma contiene un errore di sintassi. D Al punto 2: A[i-1] ha valore 0.0. E Il programma stampa "Ciao".</p> |
| <p>7. [3.5]</p> | <p>Si considerino i vettori nel linguaggio C:</p> <p>A L'indice di un elemento puo' essere una variabile di tipo float. B L'indice di un elemento di un vettore deve essere minore della sua dimensione logica. C Non e' possibile applicare l'operatore di assegnamento a variabili di tipo vettore. D Non e' possibile stampare il contenuto di un vettore di 3 interi con una sola printf. E Gli elementi di un vettore sono tutti dello stesso tipo.</p> | <p>Si considerino i tipi di dato scalari primitivi nel linguaggio C:</p> <p>A La sottrazione tra un float e un char produce un valore di tipo float. B Il criterio di equivalenza strutturale puo' consentire l'assegnamento del valore di una variabile di tipo non primitivo a una variabile di tipo scalare primitivo. C L'<i>overloading</i> degli operatori permette di elaborare con lo stesso operatore tipi di dati altrimenti incompatibili . D L'<i>overloading</i> degli operatori permette di denotare operatori diversi con lo stesso simbolo. E E' possibile eseguire una somma tra una variabile float e una variabile char</p> <p>8. [3.5]</p> |
| <p>9. [3.5]</p> | <p>Si considerino le istruzioni del linguaggio C:</p> <p>A Nell'istruzione for il blocco di istruzioni puo' non essere mai eseguito. B Ogni istruzione switch e' sempre esprimibile mediante altre istruzioni. C Nell'istruzione for l'espressione di modifica viene eseguita nel momento in cui la condizione e' falsa. D Nell'istruzione do..while se la condizione e' verificata, si esce dal ciclo. E Nell'istruzione switch almeno un blocco di istruzioni è sempre eseguito.</p> | |