



<p>1. [4]</p>	<p>Si considerino le funzioni nel linguaggio C:</p> <p>A [F] Un file sorgente C, per essere compilato correttamente, deve contenere tutte le definizioni di tutte le funzioni che utilizza.</p> <p>B [F] Una funzione non può restituire un tipo struct.</p> <p>C [V] I parametri formali di una istanza di una funzione vengono inizializzati al valore dei parametri effettivi (attuali) all'atto della chiamata.</p> <p>D [F] Si consideri una variabile p, di tipo puntatore a int, e una funzione $\text{int } f(\text{int } *Q)$, al cui interno Q viene modificato. La chiamata $f(p)$; modifica il valore di p.</p> <p>E [F] Le istanze di funzioni utilizzano dei record di attivazione che vengono deallocati ogni volta che vengono generate ricorsivamente altre istanze della stessa funzione.</p>	<p>Si consideri il seguente programma C:</p> <pre>#include <stdio.h> #define N 9 int f(int *a, int b) { if(*a < *(a+b)) /* punto 1 */ return b; else return *a=f(a, b+1)+b; } main() { int i, V[N]={0,1,2,3,4,5,6,7,8}; for (i=N-1; i>=0; i-=2) V[i]=(i+5)/2; /*punto 2 */ printf("%d\n\n", f(V,0)); /* punto 3 */ for (i=0; i<N; i++) /* punto 4*/ printf("%d\n", V[i]); } 3. [7]</pre>
<p>2. [4]</p>	<p>Si considerino le caratteristiche del linguaggio C:</p> <p>A [F] È possibile che in un programma vengano definite due variabili diverse aventi lo stesso identificatore, ed entrambe allocate nel data segment.</p> <p>B [F] Lo stack può contenere variabili allocate dinamicamente (tramite la malloc).</p> <p>C [V] Gli identificatori stdin, stdout e stderr denotano puntatori a FILE.</p> <p>D [F] All'interno di una funzione è possibile definire blocchi di istruzioni contenenti nuove variabili, il cui campo d'azione diventa l'intero programma.</p> <p>E [F] L'operatore di assegnamento tra variabili di tipo vettore è equivalente all'operatore di assegnamento tra variabili di tipo puntatore.</p>	<p>A [F] L'istruzione prima del punto 2 ($V[i]=(i+5)/2$) viene eseguita 4 volte.</p> <p>B [F] La funzione f è tail-ricorsiva.</p> <p>C [V] L'istruzione prima del punto 1 ($\text{if}(*a < *(a+b))$) viene eseguita 3 volte.</p> <p>D [V] L'istruzione al punto 3 stampa il valore 3.</p> <p>E [V] Al punto 1 la variabile V non è visibile.</p>