

<p>1. [4]</p>	<p>Si considerino le funzioni nel linguaggio C:</p> <p>A [F] Il valore restituito da una funzione ricorsiva è sempre ottenuto come risultato di una ulteriore chiamata ricorsiva a funzione.</p> <p>B [V] Il corpo di una funzione ricorsiva f può contenere anche due o più chiamate a f.</p> <p>C [V] Si consideri una variabile p, di tipo puntatore a int, e una funzione int f(int *Q), al cui interno Q viene modificato. La chiamata f(p); non modifica il valore di p.</p> <p>D [V] Due istanze di una stessa funzione fanno riferimento allo stesso codice.</p> <p>E [V] I parametri formali di una funzione possono essere modificati all'interno della funzione stessa.</p>	<p>Si consideri il seguente programma C:</p> <pre>#include <stdio.h> #define K 9 int f(int *x, int y) { if(*x < *(x+y)) /* punto 1 */ return y; else return *x=f(x,y+1)+y; } main() { int i, A[K]={0,1,2,3,4,5,6,7,8}; for(i=K-1; i>=0; i-=2) A[i]=(i+5)/2; /*punto 2 */ printf("%d\n\n", f(A,0)); /* punto 3 */ for(i=0; i<K; i++) /* punto 4*/ printf("%d\n", A[i]); } A [F] L'istruzione al punto 1 (if(*x<*(x+y))) viene eseguita 1 volta. B [F] Al punto 3 la variabile i ha valore 2. C [F] L'istruzione al punto 3 (printf) stampa il valore 6. D [F] La funzione f è ricorsiva "tail". E [V] La funzione f genera degli effetti collaterali.</pre>
<p>2. [4]</p>	<p>Si considerino le caratteristiche del linguaggio C:</p> <p>A [V] Tramite il puntatore a FILE è possibile accedere a un file binario per leggere o scrivere dei record.</p> <p>B [F] Tutte le variabili di un programma C sono contenute nello stack.</p> <p>C [V] È possibile tramite la malloc definire vettori di lunghezza non nota al momento della compilazione.</p> <p>D [F] &p, dove p è un puntatore a intero, denota l'indirizzo dell'intero puntato da p.</p> <p>E [F] Tramite la funzione fseek() è possibile leggere una sequenza di caratteri da un file di testo.</p>	<p>3. [7]</p>