



CORREZIONE PROVA SCRITTA DI PROGRAMMAZIONE DEL 11 FEBBRAIO 2002

1. Si considerino i seguenti comandi C:

- (a) `7==7;` (b) `7=7;` (c) `7;` (d) `5=7;`

Solo i comandi (a) e (c) non provocano errori statici

Nei comandi (b) e (d) viene rilevato il tentativo di assegnare un valore a un'espressione che non è un designatore

2. Cos'è un "comando semplice"?

Un'espressione seguita dal carattere ';'.

Un comando di assegnamento è un particolare comando semplice. Un comando semplice come il comando "7;" dell'esercizio precedente non modifica lo stato della memoria.

3. Quali delle seguenti espressioni è falsa?

Un'espressione di tipo T può essere assegnata solamente ad una variabile di tipo T.

Ad esempio possiamo assegnare un valore di tipo char ad una variabile di tipo int.

4. Che cosa produce in output la seguente porzione di codice?

```
int x, y=2;
for (x=1, y*=x; x > --y; x+=2, printf("(%d,%d) ", x, y)) {
    printf("(%d,%d) ", --x, y); }
printf("FINE") ;
```

FINE

La condizione `x > --y` non è verificata, quindi dopo le istruzioni di inizializzazione si esce subito dal ciclo for.

5. Le variabili locali dichiarate all'interno di una procedura

Sono allocate nello stack

6. Quale delle seguenti espressioni booleane testa correttamente se due variabili intere x e y hanno lo stesso segno?

`(x*y >=0)`

`(x>=0) && (y>=0) || (x<=0) && (y<=0)`

Entrambe le risposte erano corrette.

7. Completare la definizione della seguente procedura *ricorsiva* `scrivi(int n)` che dato numero intero *negativo* n scrive su una stessa riga i numeri da n a 0.

**`void scrivi(int n){`
 `if (n<=0){ printf("%d", n); scrivi(n+1); }`
 `}`**

8. Completare la seguente funzione in modo che restituisca la somma degli elementi dell'array bidimensionale di interi A di dimensioni N*M (dove N e M sono costanti).

```
❑ int add(int A[N][M]) {
    int i,j, risultato;
    for(i=0, risultato=0; i<N; i++)
        for(j=0; j<M; j++)
            risultato+=A[i][j];
    return risultato;
}
```

9. NULL...

❑ È un valore che appartiene sia ai tipi puntatore che ai tipi procedura

10. La portata (o campo di validità) di un identificatore...

❑ Si può determinare direttamente dal programma (codice)

11. Quale delle seguenti espressioni è falsa?

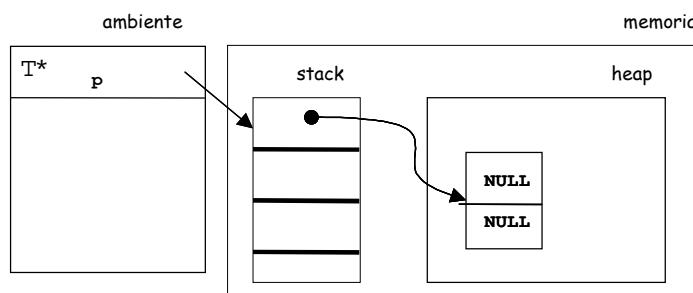
❑ Un valore riferimento non può essere passato come parametro attuale ad una procedura.

12. Si considerino le seguenti dichiarazioni di tipo:

```
typedef struct Lista{ int val;
                    struct Lista* next;
} Lista;

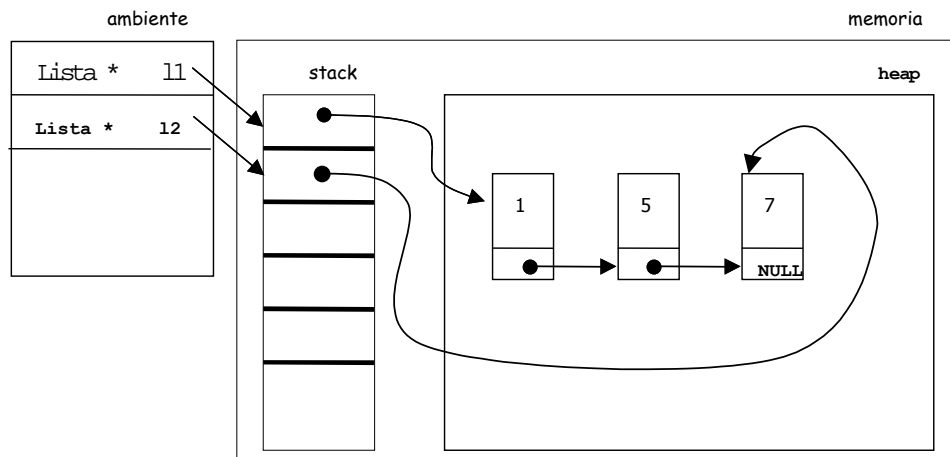
typedef struct T{
    struct Lista* primo;
    struct Lista* secondo;
} T;
```

Scrivere i comandi necessari perché si realizzi il seguente stato nel modello ambiente-memoria:



```
T* p = (T*) malloc(sizeof(T));
p->primo = NULL;
p->secondo = NULL;
```

13. Utilizzando il tipo Lista definito nell'esercizio precedente, scrivere i comandi necessari perché si realizzi il seguente stato nel modello ambiente-memoria.

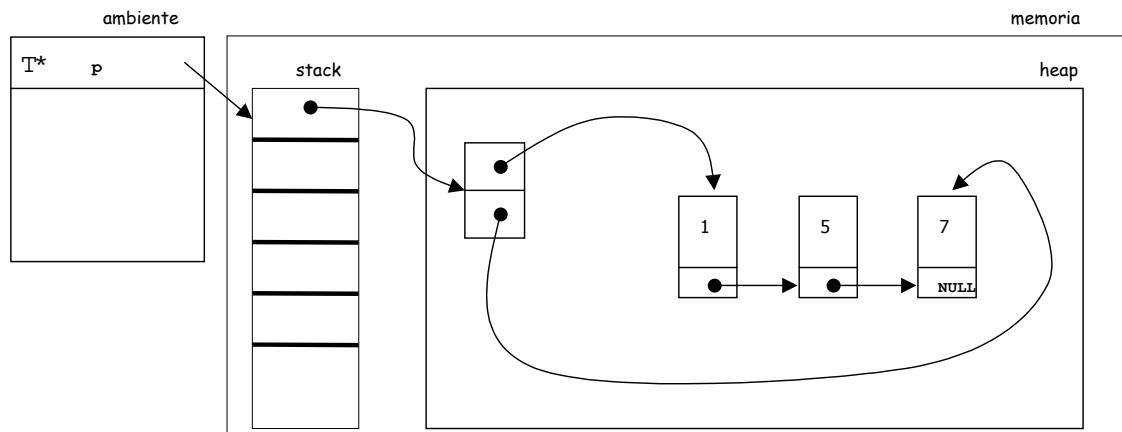


```

□ Lista l1, l2;
l2= (Lista*) malloc(sizeof(Lista));
l2->val=7;
l2->next = NULL;
l1= (Lista*) malloc(sizeof(Lista));
l1->val=1;
l2->next=(Lista*) malloc(sizeof(Lista));
l1->next->val=5;
l1->next->next=l2;

```

14. Utilizzando i tipi definiti nell'esercizio 12, scrivere i comandi necessari perché si realizzi il seguente stato nel modello ambiente-memoria.



```

T* p = (T*) malloc(sizeof(T));
p->secondo = (Lista*) malloc(sizeof(Lista));
p->secondo->val=7;
p->secondo->next=NULL;
p->primo = (Lista*) malloc(sizeof(Lista));
p->primo->val=1;
p->primo->next=(Lista*) malloc(sizeof(Lista));
p->primo->next->next=p->secondo ;

```

15. Quale delle seguenti espressioni è falsa, per il linguaggio C ?

- E' possibile durante la computazione costruire un valore di tipo procedura.**

Questa è una caratteristica dei cosiddetti "linguaggi funzionali"

16. E' vero o falso che quando una procedura viene chiamata vengono introdotte nell'ambiente della procedura, oltre alle variabili locali dichiarate all'interno della stessa, anche una batteria di variabili locali in corrispondenza dei suoi parametri formali ?

- Vero**
-