

Nome e Cognome: _____

Matricola: _____

- 5 punti** Descrivere la regola generale di conversione per un'espressione con due valori di tipo intero x operazione y in caso x abbia tipo *signed TipoT* ed un grado di conversione più elevato di quello dell'altro operando (y),
- 5 punti** Scrivere cosa stampa la seguente porzione di codice.

```

1 int a= 0xfe - 0403;
2 while (!(a++ && a++, --a), !!a) {
3     printf("%d \n", a);
4     if (a + 2 >= -0x2) {
5         a + 1;
6         continue; break;
7     }
8     printf("CIAO\n");
9     break;
10 }
11 printf("a: %d\n", a - ? 1 : -1);

```

```

-4
-3
-2
-1
a: -1

```

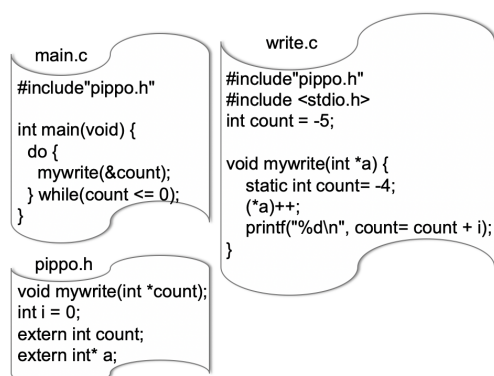
- 6 punti** Data la seguente struttura che definisce un nodo della lista globale *pFirst*, definire una funzione di nome *del_add_head* che prende come parametro *int infoToRemove*, e rimuove l'elemento in testa alla lista solamente se il campo *info* di questo elemento ha valore minore o uguale a *infoToRemove*. In caso contrario, la funzione aggiunge in testa un nuovo elemento della lista con campo *info* uguale a *infoToRemove*.

```

1 struct Node {
2     int info;
3     struct Node* pNext;
4 }

```

- 7 punti** Dire quali compilazioni provocano errore a causa del linker (e perché): 1) *gcc -o write write.c*, 2) *gcc -c main.c*, 3) *gcc -o main main.c*, 4) *gcc write.c main.c -o output*. In caso il punto 4) ritorni un errore, descrivere come può essere corretto. Dopo aver corretto l'eventuale errore, che *linkage* hanno *count*, *i*, *a*, e *mywrite*? Cosa stampa il programma *output*? Elencare tutte le definizioni e dichiarazioni in ogni file.



1) Errore, manca main - 3) Errore manca mywrite e count
4) Errore, simbolo *i* definito più volte (in *main.c* e *write.c* perché importato con *include* in *pippo.h*). Una possibile soluzione consiste nell'eliminare tutta l'assegnamento a zero di in *pippo.h* lasciando "int i;"

- count ha linkage esterno in <i>main.c</i> , esterno e nolinkage in <i>write.c</i> (ci sono due variabili con lo stesso nome <i>count</i>)	Output: -4
- <i>i</i> ha linkage esterno in <i>main.c</i> e <i>write.c</i>	-4
- <i>a</i> ha linkage esterno in <i>main.c</i> e <i>write.c</i>	-4
- <i>mywrite</i> ha linkage esterno in <i>main.c</i> e <i>write.c</i>	-4
<i>main.c</i> : <i>main</i> definito	-4
<i>write.c</i> : <i>count</i> definito due volte, <i>mywrite</i> definito, <i>a</i> definito	-4

pippo.h: *count*, *a*, *i*, *mywrite* dichiarati

- 7 punti** Cerchiare le affermazioni vere dato $\text{int } a[5] = \{129, INT_MIN, INT_MIN \mid INT_MAX, 262142, 262168\}$; $\text{short int } *p = (\text{short} *) a$; $\text{char } *q = (\text{char} *) a$; $q[1] = 1$; sapendo che i tre tipi usati occupano 4, 2, e 1 byte, e $262144 = 2^{18}$ (valori rappresentati in *little endian* e complemento a due). Scrivere la mappa di memoria e giustificare le affermazioni (vere o false). A. $\&a[5] - (a + 2) - q[1] - 2$ B. $*(p + 2) ? p[7] : !p[4] + p[5] + p[7]$ C. $*((\text{short}*)(q + 13)) == *((\text{short}*)(q + 17))$

Esercizio 1

x op y (slide su conversione tipo)

Se x ha tipo con segno signed TipoT il cui grado di conversione è più elevato di quello dell'altro operando (y), si applica questa regola. L'altro operando (y) è convertito a signed TipoT solo se questo tipo è in grado di rappresentare tutti i valori di y. Altrimenti, tutti e due gli operandi (x e y) sono convertiti a unsigned TipoT.

Esercizio 5

tutti i 3 punti sono falsi

Esercizio 3

```
void del_head_add_head(int infoToRemove) {
    if (pFirst == NULL)
        printf("Nessun elemento da cancellare!");
    }
    else {
        if (pFirst->info <= infoToRemove) {
            Node* temp= pFirst->pNext;
            free(pFirst);
            pFirst= temp;
        }
        else {
            Node *pNew= (Node*) malloc(sizeof(Node));
            pNew->info= infoToRemove;
            pNew->pNext= pFirst;
            pFirst= pNew;
        }
    }
    return;
}
```

10000000 1 $\leftarrow q[1]$
 (10000000 0) $\leftarrow q[1]$

00 00000 0
 00 00000 0

(0 000000 0) $\leftarrow a+1$
 (0 0 00000 0) $\leftarrow *(p+2)$

00 00000 0
 00 00000 1

11 111111 1 $\leftarrow a+2$
 { 11 111111 1 } $\leftarrow a[2]$
 { 11 111111 1 } $\leftarrow *(p+5) == p[5]$
 (11 111111 1)
 (11 111111 1)

(0 11111 11) $\leftarrow p[6]$
 (1 11111 11) $\leftarrow *(sum+)(q+13)$
 (1 1 00000 0) $\leftarrow p[7]$
 (0 000000 0)

(00011000) $\leftarrow q[16]$
 (00000000) $\leftarrow ((sum+)(q+17))$
 (00100000) $\leftarrow q[18]$
 (00000000)

x x x x x x x x $\leftarrow da[5]$
 x x x x x x x x $\leftarrow p+10$
 x x x x x x x x
 x x x x x x x x

A

$$\frac{da[5] - (a+2)}{3} - q[1] - 2 == 0$$

- 1 FALSA

B

$$*((sum+)(q+13)) == *((sum+)(q+17))$$

1023 == 1024

FALSA

$$*(p+2) == 0 \quad C$$

avinoi

$$!(p[4] + p[5] + p[7])$$

$$!(-1 + (-1) + 3) == !1 == 0$$

FALSA