

Nome e Cognome: _____

Matricola: _____

1. 3 punti Elencare tutte le conversioni di tipo presenti, implicite ed esplicite: *da tipo1 a tipo2*.

```
1 char a = '?';
2 unsigned int b = 71 + a;
3 int c = 2.5 + 0x12;
4 float x = c / a;
```

a da char a long int / '?' da int a char
 71 da long int a unsigned int
 0x12 da int a double
 2.5 da double a int
~~c da char a int~~, c/a da int a float
 a da char a int

2. 3 punti i) Elencare tutte le conversioni di tipo come nel primo esercizio, e ii) scrivere il valore finale di *x*.
 Suggerimento: -1 è rappresentato in memoria come una sequenza di 32 bit con valore 1.

```
1 int i = -1;
2 unsigned int limit = 0xc8;
3 long n = 0x1e;
4 float x = 0.0;
5 if (i < limit)
6     x = limit * n;
7 else
8     x += 2.5;
```

0xc8 da int a unsigned int
 0x1e da int a long int
 0.0 da double a float
 -1 da int a unsigned int ($= 2^{32} - 1 > \text{LIMIT!}$)
 (x = LIMIT * n non eseguito) x = 2.5

3. 3 punti Scrivere cosa stampa il seguente programma.

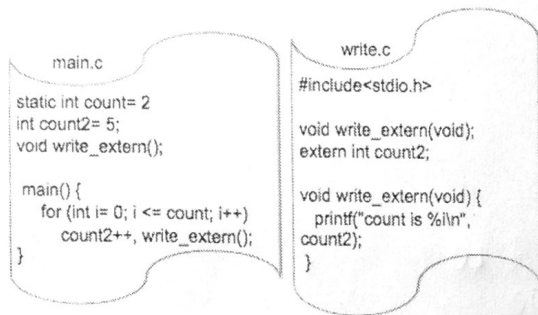
```
1 #include <stdio.h>
2
3 int main (void) {
4     int i, w = 5;
5     for (i=1; i<= w*(w+1); i++)
6         printf ("%c", i % (w+1) == 0 ? '\n' : (i % (w) == 0 || i % (w+2) == 1) ? '*' : '-');
7 }
```

```
* - - - *
- * - * -
- - * - -
- * - * -
* - - - *
```

4. 4 punti Domanda aperta: riportare le differenze tra dichiarazione e definizione di variabili e funzioni. Elen-
 care le regole che permettono di individuare se dichiarazione o definizione. Per esempio, "se uso lo specificatore
extern su una variabile e non la assegno, allora la variabile è...".

CAPITULO 4 LIBRO (TYPE CONVERSIONS)

5. 3 punti Dire quali dei seguenti comandi di compilazione ritorna un errore a causa del linker (e perché): 1) *gcc -o main main.c*, 2) *gcc -c write.c*, 3) *gcc -o write write.c*, 4) *gcc -o main write.c main.c*. Inoltre, riportare
 l'output finale del programma e, per ogni identificatore, il tipo di linkage, e in quale file è definito o dichiarato.



1) errore (WRITE_EXTERN NON DEFINITA)
 2) OK, 3) errore (COUNT2, MAIN NON DEFINITE)
 4) OK.

WRITE_EXTERN DICHIARATO IN WRITE.C
 COUNT2 DICHIARATO IN WRITE.C
 WRITE_EXTERN DEFINITO IN WRITE.C
 I DEFINITO IN MAIN.C, NO LINKAGE

COUNT DEFINITO IN MAIN.C, LINKAGE INTERNO
 COUNT2 DEFINITO IN MAIN.C, LINKAGE ESTERNO
 WRITE_EXTERN DICHIARATO IN MAIN.C (LINK EXT)

6. **3 punti** Cosa stampa il programma? Gli operatori unari hanno precedenza massima, a si trova all'indirizzo di memoria `0x7fff54824ffa`, uno `short` occupa 2 byte, un `int` 4 byte, un `long` 8 byte. Suporre anche che, se l'operando sinistro di un operatore `||` è vero, l'operando destro NON viene valutato.

```

1 int a = 3, *b = &a;
2 int c = !(a--=1, ((a==6) || ++a));
3 int d = !c && (a--=1, (a==6) && ++a);
4 printf("%d %d %d\n", a, c, d);
5 printf("%p %p %lu\n", b, (long *) (
    short *) b + 1, sizeof(*b));

```

-10 0 1
 0x7fff54824ffa 0x7fff54825002/4

7. **4 punti** Scrivere cosa stampa il seguente programma.

```

1 int f(int* b, int c, int* d) {
2     static int a = 1;
3     printf("%d \n", *d*c);
4     return (a++ * (*b)++ * c - * (*d)
5         )++);
6 }
7 int main(void) {
8     int a = 2, b = 3, c = 1;
9     b = f(&a, b, &c);
10    printf("%d %d %d\n", a, b, c);
11    b = f(&a, b, &c);
12    printf("%d %d %d\n", a, b, c);
13    b = f(&c, -3, &a);
14    printf("%d %d %d\n", c, b, a);
15 }

```

3
 3 6 2
 12
 4 72 3
 -12
 4 -108 5

8. **3 punti** Scrivere una funzione ricorsiva che, preso un valore n di tipo `int`, ritorna la somma dei primi n valori.

```

INT ADDNUMBERS (INT N) {
    IF (N > 1)
        RETURN N + ADDNUMBERS (N-1);
    ELSE
        RETURN N;
}

```

9. **4 punti** Cerchiare le affermazioni vere dato `int a[4] = {1, 0, 1}`; `short int *p = (short int*) a`; sapendo che un `short/int/long` occupa 2/4/8 byte, e $65536 = 2^{16}$ (valori interi rappresentati in *little endian*).

A) $*(a+2)$ vale 0; B) $p[2]$ vale 65536; C) $((int*)(p+1)) > 65536$; D) $((long*)a+2) >= 65537$; E) $\&a[2] >= ((int*)p+5)$; F) $((int)(p+3)) - (int)(\&a[1]) == 2$.