Prova scritta Programmazione Procedurale con Lab. - 5 Settembre 2023

Nome e Cognome: _____ Matricola: _____

1. [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*),

2. [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
```

3. [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 }
```

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.

main.c
```
#include"pippo.h"

int main(void) {
  do {
    mywrite(&count);
  } while(count <= 0);
}
```

write.c
```
#include"pippo.h"
#include <stdio.h>
int count = -5;

void mywrite(int *a) {
   static int count= -4;
   (*a)++;
   printf("%d\n", count= count + i);
}
```

pippo.h
```
void mywrite(int *count);
int i = 0;
extern int count;
extern int* a;
```

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;"

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

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

5. [7 punti] Cerchiare le affermazioni vere dato *int a[5]= {129, INT_MIN, INT_MIN | INT_MAX, 262142, 262168}; short int *p = (short*) a; char *q= (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. $^*((short^*)(q + 13)) == ^*((short^*)(q + 17))$

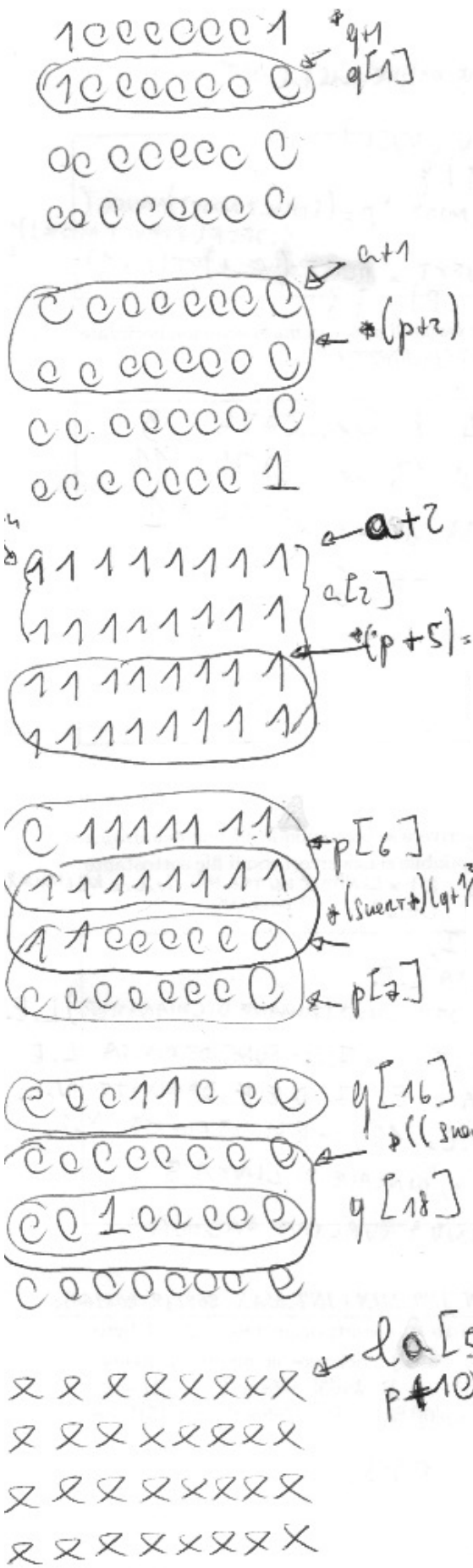
Esercizio 1
x op y (slide su conversione tipo)
Se x ha tipo con segno signed TipoT il cui grado di conversione è più elevato di quello dell'altro operando (y), si applica questa regola. L'altro operando (y) è convertito a signed TipoT solo se questo tipo è in grado di rappresentare tutti i valori di 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;
}
```



A

$$\underbrace{la[5] - (a+2)}_{3} - q[1] - 2 == 0$$

$$3 - 1$$

FALSA

B

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

1023 == 1024

FALSA

C

$*(p+2) == 0$

QUINDI

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

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

FALSA