

Matr.: \_\_\_\_\_ Nome: \_\_\_\_\_ Cognome: \_\_\_\_\_

Architettura degli Elaboratori  
prova scritta di Assembly

ISTRUZIONI:

1. Scrivete subito il vostro nome, cognome e numero di matricola su questo foglio.
2. Aprite un editor a vostra scelta, scrivete una prima linea di commento con le stesse informazioni formattate nel seguente modo: `#matricola nome cognome`  
poi salvate il file chiamandolo: `cognomen.s` (con il vostro cognome e iniziale del nome!).
3. Tutti gli esercizi vanno scritti su questo file.
4. Per consegnare, bisogna risolvere i primi due esercizi e verificare con `xspim` che la soluzione sia corretta con i dati dell'esempio.
5. L'etichetta **main** va messa davanti alla funzione di test dell'esercizio 4 se ci arrivate e funziona altrimenti dovete metterla davanti al primo test (esercizio 2).
6. Chi consegna e ha rispettato quanto scritto sopra prende almeno 3 punti (la sufficienza).
7. In ogni caso, prima di uscire dovete restituirmi questo foglio e dirmi se consegnate oppure no.

**Esercizio 1 (3 punti)**

Scrivere una funzione **trovaProdotto** che prende come parametri un puntatore ad un *menu*, e un codice prodotto (16 bit senza segno). Il *menu* è un array zero terminato di puntatori a prodotti. Un *prodotto* è una coppia di interi a 16 bit senza segno: il codice e il prezzo. La funzione restituisce il puntatore al prodotto con il codice richiesto; se manca restituisce 0. Ad esempio, dato il codice prodotto 3 e il menu: p1,p2,p3 contenente i prodotti: p1 = 13,5 p2 = 3,1 e p3 = 8,6; la funzione restituisce il puntatore: p2.

**Esercizio 2 (1 punto)**

Scrivere una funzione **trovaProdottoTest** senza parametri e con una sezione dati contenente i dati dell'esempio dell'esercizio 1. Il test applica la funzione **trovaProdotto** ai dati e stampa il prezzo del prodotto trovato (altrimenti niente).

**Esercizio 3 (3 punti)**

Scrivere una funzione **conto** che prende come parametri un puntatore al menu e un puntatore ad un array zero terminato di codici di prodotti. La funzione restituisce il prezzo totale di tutti i codici (prodotti) presenti nell'array. La funzione deve chiamare **trovaProdotto** e ignorare eventuali prodotti non presenti in menu. Ad esempio, dati il menu dell'esercizio 1 e l'array di codici: 3,9,13,3; la funzione restituisce: 7 (1+0+5+1).

**Esercizio 4 (1 punto)**

Scrivere una funzione **contoTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. La funzione chiama **conto** e stampa il risultato.

NB. Si ricorda che per stampare un intero lo si deve caricare in `$a0`, caricare 1 in `$v0` e chiamare `syscall`; invece per stampare una stringa la si deve caricare in `$a0`, caricare 4 in `$v0` e chiamare `syscall`.

Matr.: \_\_\_\_\_ Nome: \_\_\_\_\_ Cognome: \_\_\_\_\_

Architettura degli Elaboratori  
prova scritta di Assembly

ISTRUZIONI:

8. Scrivete subito il vostro nome, cognome e numero di matricola su questo foglio.
9. Aprite un editor a vostra scelta, scrivete una prima linea di commento con le stesse informazioni formattate nel seguente modo: `#matricola nome cognome`  
poi salvate il file chiamandolo: `cognomen.s` (con il vostro cognome e iniziale del nome!).
10. Tutti gli esercizi vanno scritti su questo file.
11. Per consegnare, bisogna risolvere i primi due esercizi e verificare con `xspim` che la soluzione sia corretta con i dati dell'esempio.
12. L'etichetta **main** va messa davanti alla funzione di test dell'esercizio 4 se ci arrivate e funziona altrimenti dovete metterla davanti al primo test (esercizio 2).
13. Chi consegna e ha rispettato quanto scritto sopra prende almeno 3 punti (la sufficienza).
14. In ogni caso, prima di uscire dovete restituirmi questo foglio e dirmi se consegnate oppure no.

**Esercizio 1 (3 punti)**

Scrivere una funzione **addIngrediente** che prende come parametri un puntatore ad una *ricetta*, un codice ingrediente e una quantità (entrambi a 8 bit senza segno). La *ricetta* è un array zero terminato di puntatori a ingredienti. Un *ingrediente* è una coppia di interi a 8 bit senza segno: il codice e la quantità. La funzione aggiunge la quantità richiesta dell'ingrediente. Si assume che l'ingrediente sia presente. Ad esempio, dato il codice ingrediente 3, la quantità 4 e la ricetta: `i1,i2,i3` contenente gli ingredienti: `i1 = 2,1` `i2 = 3,5` e `i3 = 5,6`; la funzione modifica l'ingrediente: `i2 = 3,9`.

**Esercizio 2 (1 punto)**

Scrivere una funzione **addIngredienteTest** senza parametri e con una sezione dati contenente i dati dell'esempio dell'esercizio 1. Il test applica la funzione **addIngrediente** ai dati e stampa la quantità del secondo ingrediente.

**Esercizio 3 (3 punti)**

Scrivere una funzione **addTorta** che prende come parametri due puntatori a ricette. La funzione modifica la prima ricetta aggiungendo le quantità di ingredienti richieste dalla seconda. La funzione deve chiamare **addIngrediente**. Ad esempio, data la ricetta dell'esercizio 1 e la ricetta: `i4,i5` contenente gli ingredienti: `i4 = 2,3` `i5 = 5,1`; la funzione modifica gli ingredienti della prima ricetta come segue: `i1 = 2,4` (`1+3`) e `i3 = 5,7` (`6+1`) con `i2` che resta invariato.

**Esercizio 4 (1 punto)**

Scrivere una funzione **addTortaTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. La funzione chiama **addTorta** e stampa tutte le quantità di ingredienti della prima ricetta.

NB. Si ricorda che per stampare un intero lo si deve caricare in `$a0`, caricare 1 in `$v0` e chiamare `syscall`; invece per stampare una stringa la si deve caricare in `$a0`, caricare 4 in `$v0` e chiamare `syscall`.