

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 **appendiMinori** che prende come parametri un puntatore a stringa, un carattere e un puntatore ad un buffer. La funzione copia nel buffer tutti i caratteri della stringa minori (ascii) del carattere dato. Il buffer deve essere zero terminato. Ad esempio, dati: "assembly" e 'm'; la funzione scrive nel buffer: "aembl".

Esercizio 2 (1 punto)

Scrivere una funzione **appendiMinoriTest** senza parametri e con una sezione dati contenente i dati dell'esempio dell'esercizio 1. Il test applica la funzione **appendiMinori** ai dati e stampa il buffer.

Esercizio 3 (3 punti)

Scrivere una funzione **appendiMinoriStringa** che prende come parametri due puntatori a stringhe e un buffer. La funzione chiama **appendiMinori** per ogni carattere della seconda stringa passando sempre la prima stringa e facendo in modo che tutti i risultati vengano scritti di seguito in uno stesso buffer. Ad esempio, dati: "assembly" e "fun"; la funzione scrive nel buffer: "aebassemblaembl".

Esercizio 4 (1 punto)

Scrivere una funzione **appendiMinoriStringaTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. La funzione chiama **appendiMinoriStringa** e stampa il buffer.

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 **alfabetoUsato** che prende come parametri un puntatore a stringa e un puntatore a un buffer. La funzione scrive nel buffer le lettere della stringa ciascuna nella posizione corrispondente al proprio codice ascii facendo in modo che la lettera 'a' finisca nella prima posizione del buffer e le altre a seguire. Ad esempio, data la stringa: "caffè", la funzione scrive nel buffer: "a_c_ef_____" (dove '_' sta per un carattere già presente nel buffer e che non viene modificato da questa funzione).

Esercizio 2 (1 punto)

Scrivere una funzione **alfabetoUsatoTest** senza parametri e con una sezione dati contenente i dati dell'esempio dell'esercizio 1 (per il buffer usare `.space`). Il test applica la funzione **alfabetoUsato** ai dati e stampa il buffer (come sequenza di interi a 8 bit della lunghezza allocata).

Esercizio 3 (3 punti)

Scrivere una funzione **stringaAlfabetoUsato** che prende come parametri un puntatore a stringa e un puntatore a un buffer. La funzione inizializza il buffer con 32 '_' seguiti da uno zero e chiama **alfabetoUsato**. La funzione restituisce il puntatore al buffer. Ad esempio, data la stringa: "caffè" e un buffer contenente "blablablablablablabla..."; la funzione restituisce un puntatore alla stringa "a_c_ef_____".

Esercizio 4 (1 punto)

Scrivere una funzione **stringaAlfabetoUsatoTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. La funzione chiama **stringaAlfabetoUsato** e stampa il buffer (come stringa).

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`.