

Nome: _____ Cognome: _____ Matr.: _____

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 (2 punti)

Scrivere una funzione **copiaOccorrenze** con tre parametri: un puntatore ad una stringa, un puntatore ad un buffer e un carattere. La funzione copia il carattere nel buffer tante volte quante sono le sue occorrenze nella stringa. Il buffer deve essere terminato con zero. Ad esempio, data la stringa: "trattore" e il carattere 't'; la funzione scrive nel buffer: "ttt".

Esercizio 2 (2 punti)

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

Esercizio 3 (3 punti)

Scrivere una funzione **graficoOccorrenze** che prende come parametri due puntatori a stringa: *testo*, *pattern*. La funzione per ogni carattere del *pattern* chiama **copiaOccorrenze** sulla stringa *testo* e stampa il buffer risultante. Ad esempio, date le stringhe: "trattore", "tar"; la funzione stampa: "ttt", "a", "rr".

Esercizio 4 (1 punto)

Scrivere una funzione **graficoOccorrenzeTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. Il test applica la funzione **graficoOccorrenze** ai dati.

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`.

Nome: _____ Cognome: _____ Matr.: _____

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 (2 punti)

Scrivere una funzione **filtraOccorrenze** con tre parametri: un puntatore ad una stringa, un puntatore ad un buffer e un carattere *c*. La funzione copia nel buffer tutti i caratteri della stringa ad eccezione di *c*. Il buffer deve essere terminato con zero. Ad esempio, data la stringa: "aratro" e il carattere 'r'; la funzione scrive nel buffer: "aato".

Esercizio 2 (2 punti)

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

Esercizio 3 (3 punti)

Scrivere una funzione **mostraConFiltri** che prende come parametri due puntatori a stringa: *testo*, *filtri*. La funzione per ogni carattere della stringa *filtri* chiama **filtraOccorrenze** sulla stringa *testo* e stampa il buffer risultante. Ad esempio, date le stringhe: "aratro", "tar"; la funzione stampa: "araro", "rtro", "aato".

Esercizio 4 (1 punto)

Scrivere una funzione **mostraConFiltriTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. Il test applica la funzione **mostraConFiltri** ai dati.

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`.

Nome: _____ Cognome: _____ Matr.: _____

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 (2 punti)

Scrivere una funzione **contaOccorrenze** che prende come parametri un puntatore ad un array di interi a 16 bit senza segno, la sua lunghezza (intero a 8 bit) e un intero a 16 bit senza segno. La funzione restituisce il numero di occorrenze dell'intero nell'array. Ad esempio, dato l'array: 2,4,4,3,2,3,4 di lunghezza 7 e l'intero: 4; la funzione ritorna 3.

Esercizio 2 (2 punti)

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

Esercizio 3 (3 punti)

Scrivere una funzione **maxOccorrenze** che prende come parametri un puntatore ad un array di interi a 16 bit senza segno e la sua lunghezza (intero a 8 bit). La funzione chiama **contaOccorrenze** passandogli l'array, la sua lunghezza e ciascun intero dell'array stesso e restituisce il risultato massimo. Ad esempio, dato l'array: 2,4,4,3,2,3,4 di lunghezza 7; la funzione restituisce 3.

Esercizio 4 (1 punto)

Scrivere una funzione **maxOccorrenzeTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. La funzione chiama **maxOccorrenze** 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`.

Nome: _____ Cognome: _____ Matr.: _____

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 (2 punti)

Scrivere una funzione **decrementaOccorrenze** che prende come parametri un puntatore ad un array di interi a 8 bit senza segno, la sua lunghezza (intero a 8 bit) e un intero a 8 bit senza segno. La funzione modifica l'array sottraendo 1 a ciascuna occorrenza dell'intero e non restituisce niente. Ad esempio, dato l'array: 2,4,4,3,2,3,4 di lunghezza 7 e l'intero: 4; la funzione modifica l'array come segue: 2,3,3,3,2,3,3.

Esercizio 2 (2 punti)

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

Esercizio 3 (3 punti)

Scrivere una funzione **azzerxFino** che prende come parametri un puntatore ad un array di interi a 8 bit senza segno, la sua lunghezza (intero a 8 bit) e un intero a 8 bit senza segno. La funzione chiama **decrementaOccorrenze** passandogli l'array, la sua lunghezza e un intero che varia dal valore passato fino a zero. Ad esempio, dato l'array: 2,4,4,3,2,3,4 di lunghezza 7 e 3; la funzione modifica l'array come segue: 0,4,4,0,0,0,4.

Esercizio 4 (1 punto)

Scrivere una funzione **azzerxFinoTest** senza parametri; con una sezione dati contenente i dati dell'esercizio 3. La funzione chiama **azzerxFino** 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`.