

**Corsi di Laurea in Ingegneria Biomedica e Ingegneria Elettronica, a.a. 2007-08**  
**Modulo di “Informatica I”**

**Prova intermedia del 11 febbraio 2008 - Fila ‘A’**

**durata della prova: 2h**

**- Domande a risposta libera -**

1. Descrivere i tipici *flussi di un programma*
2. Come può avvenire la comunicazione tra un programma chiamante e un sottoprogramma?
3. Cosa si intende per *clock* di un calcolatore?
4. Descrivere le fasi del processo di compilazione.
5. Che cosa si intende con programmazione orientata agli eventi.

**- Esercizi sul linguaggio ‘C’ -**

N.B. Nel caso si ritenga necessario formulare ipotesi aggiuntive è importante indicarle chiaramente e motivarle.  
Scrivere, soprattutto il codice, in modo leggibile

1. Scrivere un programma che legga da tastiera 25 numeri reali e stampi solo quelli maggiori della loro media aritmetica.
2. Leggere da tastiera una sequenza di caratteri alfanumerici (lettere e cifre) e stampare a monitor le lettere. La sequenza deve essere interrotta quando la somma delle cifre inserite diventa superiore a 100 o quando sono stati inseriti 1000 valori.
3. Scrivere la funzione `Vocali` che stampa a monitor le vocali comprese tra i due caratteri passati come argomenti. La funzione ritorna il numero di vocali stampate. Se l'ordine dei caratteri non è corretto, la funzione deve ritornare il valore -1. Specificare il prototipo della funzione, la sua definizione ed un esempio completo di chiamata.
4. Durante un processo di produzione, un sistema di controllo registra le informazioni relative all'occorrenza di anomalie (*fault*). In particolare si registra l'ora d'inizio del *fault*, la sua durata, e un codice ('A', 'B', o 'C') che ne identifica la tipologia. Definire una struttura dati per gestire le informazioni relative a 500 *fault*. Scrivere un programma che stampi a monitor qual è la durata media dei *fault* di tipo 'A'. (Si supponga che i *fault* siano mutuamente esclusivi e che la struttura dati sia già inizializzata).
5. Dato il file testo `ex.txt`, scrivere un programma per stampare tutte le linee che superano gli 80 caratteri. (Si utilizzino le opportune funzioni di libreria contenute in `<string.h>`).

**Corsi di Laurea in Ingegneria Biomedica e Ingegneria Elettronica, a.a. 2007-08**  
**Modulo di "Informatica I"**

**Prova intermedia del 14 gennaio 2008 - Fila 'B'**

**durata della prova: 2h**

**- Domande a risposta libera -**

1. Che differenza c'è tra la memoria centrale e la memoria di massa?
2. Quali sono le funzionalità del *sistema operativo*?
3. A cosa servono i *linguaggi di programmazione*?
4. Spiegare la differenza tra file sorgente, file oggetto e file eseguibile.
5. Descrivere la struttura di un semplice programma Java.

**- Esercizi sul linguaggio 'C' -**

N.B. Nel caso si ritenga necessario formulare ipotesi aggiuntive è importante indicarle chiaramente e motivarle.  
Scrivere, soprattutto il codice, in modo leggibile.

1. Scrivere un programma che stampi in ordine inverso tutte le lettere minuscole dell'alfabeto.
2. Leggere da tastiera una sequenza di numeri reali e stampare a monitor la differenza tra un numero e il suo precedente ( $x_i - x_{i-1}$ ). La sequenza deve essere interrotta quando la differenza diventa negativa o quando sono stati inseriti 100 valori.
3. Scrivere la funzione `Pari` che stampa a monitor i numeri interi pari che sono compresi tra i due valori interi passati come argomenti e la somma dei numeri compresi nell'intervallo. I valori restituiti dalla funzione sono: -1 se vi sono numeri negativi, 0 se l'ordine degli argomenti non è corretto e 1 negli altri casi. Specificare il prototipo della funzione, la sua definizione ed un esempio completo di chiamata.
4. Durante un processo di produzione, un sistema di controllo registra le informazioni relative all'occorrenza di anomalie (*fault*). In particolare si registra l'ora d'inizio del *fault*, la sua durata, e un codice ('A', 'B', o 'C') che ne identifica la tipologia. Definire una struttura dati per gestire le informazioni relative a 500 *fault*. Scrivere un programma che stampi a monitor qual è il tipo di *fault* che accade più frequentemente. (Si supponga che i *fault* siano mutuamente esclusivi e che la struttura dati sia già inizializzata).
5. Scrivere un programma per confrontare due file testo (`uno.txt` e `due.txt`), che stampi tutte le copie di linee per le quali essi differiscono. (Si utilizzino le opportune funzioni di libreria contenute in `<string.h>`).