

PROVA INTERMEDIA DI PROGRAMMAZIONE (A)
(Programmazione Imperativa)

Cognome:

Nome:

Matricola:

Data: **02/04/2014**

Svolgere i seguenti quesiti nel foglio protocollo. Consegnare: presente testo, bella e brutta copia. Tempo previsto: 90 minuti.

Consideriamo un vettore di interi di dimensione $|Z| - |A| + 1$ con il seguente significato: un valore n in posizione i sta ad indicare che la i -ma lettera dell'alfabeto compare n volte in un testo come maiuscola o minuscola, un valore 0 sta ad indicare che non compare. Nel seguito chiamiamo questo vettore *vettore delle presenze*. Svolgere i seguenti esercizi:

1. Esercizio 1

Scrivere la funzione

```
void analizza_testo(int* vetp, char* testo);
```

che analizzi quali lettere sono presenti in `testo` e ne dia risposta in `vetp`. Si tenga presente che le lettere possono essere sia maiuscole che minuscole. Per esempio il testo 'Casacca' contiene le lettere a (3 occorrenze), c (3 occorrenze) e s (1 occorrenza). Si scriva un main di prova.

2. Esercizio 2

Scrivere la funzione

```
int lettere_presenti(char** s, int* vetp)
```

che dato un vettore delle presenze `vetp` allochi un array di char in memoria dinamica della dimensione minima necessaria per memorizzare tutte le lettere che compaiono almeno una volta rispetto alle occorrenze presenti in `vetp`. Per esempio se `vetp` ha tutti zeri tranne: `a[0]=4` e `a[3]=2`, la `lettere_presenti` istanzia un vettore di due char in cui la prima posizione è occupata dal carattere 'a' e la seconda dal carattere 'd'. La funzione ritorna 1 se l'allocazione della memoria è andata a buon fine, 0 altrimenti. Dare un main di prova che ponga attenzione alla rimozione dello spazio allocato in memoria dinamica.

3. Esercizio 3

Scrivere la funzione

```
void stampa_frequenti(int* vetp, int n)
```

che stampi le n lettere che compaiono più frequentemente nel testo. Si assuma che il valore di n sia $0 \leq n \leq |Z| - |A| + 1$, se vale 0 non si stamperà alcuna lettera. Dare un main di prova.