

Fondamenti di Informatica e Laboratorio

M. Gavanelli, E. Lamma

4 Luglio 2023

Esercizio 1 (Punti 15 su 31) (1h e 30 min)

Un file di testo, `parole.txt`, contiene parole della lingua italiana. Ciascuna parola è stringa lunga al massimo 50 char. La stessa parola può comparire più volte nel file.

Si realizzi un programma C, organizzato in **almeno** due funzioni, rispettivamente dedicate a:

1. creare una lista L in memoria centrale, ordinata sul campo parola, che riporta ciascuna parola del file una sola volta; la **funzioneA** riceve come parametri:

- un puntatore a file,
- il puntatore a L (inizializzato a `NULL` nel `main` prima della prima chiamata alla funzione),

più eventuali parametri a scelta, e restituisce il puntatore alla radice della lista L;

2. stampare su un file di uscita `uniche.txt`

- l'elenco ordinato delle parole della lista L;
- il numero totale (intero) delle parole della lista L – la stampa del totale vale 1 punto (chi non la effettua perde 1 punto sui 15 dell'esercizio).

La **funzione2** riceve come parametri:

- il puntatore a L,
- il puntatore al file di uscita,

più eventuali parametri a scelta, e restituisce `void`.

NOTA BENE: Si consegnino i sorgenti e il file di uscita generato (per A+B, consegnare i due file di uscita generati). È possibile utilizzare **librerie C** (ad esempio per le stringhe). Nel caso si strutturi a moduli l'applicazione qualunque **libreria utente** va riportata nello svolgimento.

Per chi svolge il compito A+B (Ulteriori punti 24 su tot. di 62 per A+B; tempo +30 min):

Si scriva una funzione (**funzioneAB**) che legge dal file `parole.txt` le prime 10 parole, e le inserisca in un array `V` di 10 stringhe (ognuna al massimo di 50 char). Il file `parole.txt` ha almeno 10 parole. Nel vettore `V` la stessa parola può comparire più volte.

Questa funzione (**funzioneAB**) riceve come parametri il vettore `V` e il puntatore al file `parole.txt`, più eventuali parametri a scelta, e restituisce `void`.

Nel `main`, si ordini il vettore `V` tramite una opportuna chiamata della funzione `qsort` e si stampi poi su un file di testo `outputAB.txt` il contenuto dell'array `V`.

Esercizio 2 (Punti 3 su 31) (15 min)

NOTA BENE: Si consegna la soluzione di questo esercizio in un file `complex.txt`.

Sia data la seguente funzione `p` che riceve un intero positivo e un albero binario (NON di ricerca) di interi positivi

```
int p(int i, tree T)
{ if (T!=NULL)
    if (i==T->value)
        return 1;
    else
        if (p(i,T->left))
            return 1;
        else return p(i,T->right);
    else return 0;
}
```

Si indichi cosa fa la funzione `p` e che tipo di algoritmo realizza. Se ne valuti anche la complessità asintotica come numero di test `i==T->value`, nel caso in cui il carattere `i` non appartenga all'albero `T` e l'albero abbia `N` nodi.