

# 14 Settembre 2018

---

## Esame di Fondamenti di Informatica per Ing. Gestionale

---

### Compito A - Durata 1h45'

---

### Istruzioni (leggere attentamente)

---

#### Note

La mancata osservanza delle seguenti regole può comportare la perdita di informazioni necessarie alla valutazione dell'esame.

#### Cartella di esame e registrazione dei dati dello studente

La cartella Esame contenente il compito da svolgere si trova sul Desktop. Entrare nella cartella e, prima di iniziare il compito, eseguire il programma `registrazione.pyc`. Inserire i dati personali fornendo (separatamente) *Numero di Matricola*, *Cognome* e *Nome*. Il programma genera un file `studente.txt` che non deve essere modificato manualmente. Verificare che i dati nel file `studente.txt` siano corretti; in caso di errore potete rieseguire il programma `registrazione.pyc`.

#### Svolgimento degli esercizi

Leggere attentamente il testo e risolvere gli esercizi proposti.

Per ogni esercizio avete una cartella `EsercN` che contiene un file dal nome `A_ExN.py` (dove N è il numero dell'esercizio) con lo scheletro della soluzione. Il file `A_S_ExN.pyc` contiene la soluzione precompilata dell'esercizio proposta dal docente. L'unico file da modificare è `A_ExN.py`. *Nel vostro interesse, non modificate altri file (ad esempio contenenti dati di test). Non create nuovi file.*

Per verificare la correttezza di un esercizio **DOVETE** usare il programma `TestEx.pyc`, che proverà la vostra soluzione su un certo numero di casi di test, mostrandovi il confronto tra i risultati ottenuti con la vostra soluzione e usando la soluzione proposta dal docente. Ciò ha lo scopo di facilitarvi nella ricerca e correzione di eventuale errori

E' possibile consultare la documentazione ufficiale del linguaggio Python (ad esempio premendo F1 all'interno dell'ambiente di programmazione Idle), ma non è possibile usare libri o appunti. In caso di problemi tecnici chiedere ai docenti o ai tecnici del laboratorio.

---

## Esercizi

---

1. **A\_Ex1(l,n) (9 punti)** Completare la funzione Python `A_Ex1(l,n)` che prende in ingresso una lista `l` di stringhe ed un intero non negativo `n`. La funzione deve restituire una lista contenente le stringhe costituite dagli ultimi `n` caratteri di ciascuna stringa in `l` (si può assumere che ciascuna stringa in `l` abbia un numero di caratteri non inferiore a `n`). Le stringhe nella lista restituita *devono essere ordinate alfabeticamente*. Se la lista `l` in input è vuota, la funzione deve restituire una lista vuota. Si noti anche che se `l` è non vuota e `n=0` la lista restituita contiene tutte stringhe vuote.

**Esempio:** se `l= ["prozio", "crocevia"]` e `n = 3`, la funzione dovrebbe restituire la lista `["via", "zio"]`.

2. **A\_Ex2(m) (9 punti)** Completare la funzione Python `A_Ex2(m)` che riceve in ingresso una *matrice quadrata* `m` di interi rappresentata per righe come lista di liste e restituisce una nuova matrice delle stesse dimensioni di `m` in cui l'elemento in posizione `i,j` è ottenuto moltiplicando l'elemento di `m` in posizione `i,j` con l'elemento di `m` in posizione `j,i`. Se `m` è la matrice vuota `[]`, oppure `m` non è una matrice quadrata, la funzione deve restituire la matrice vuota. **Esempio:** se la matrice fosse

$$\begin{pmatrix} 2 & 1 \\ 3 & 5 \end{pmatrix}$$

la funzione dovrebbe restituire la matrice

$$\begin{pmatrix} 4 & 3 \\ 3 & 25 \end{pmatrix}$$

se invece la matrice fosse

$$\begin{pmatrix} 2 & 1 & 4 \\ 3 & 5 & 7 \end{pmatrix}$$

la funzione dovrebbe restituire `[]`.

3. **A\_Ex3(file) (9 punti)** Implementare la funzione `A_Ex3(file)` che prende in ingresso il nome di un file, contenente i dettagli degli incassi di un supermercato nel seguente formato *"Nome del cassiere, Numero della cassa, Pezzi acquistati dal cliente, Prezzo pagato"* (ogni riga del file contiene i dettagli di un singolo incasso). La funzione deve restituire la cassa che ha ricevuto, complessivamente, l'incasso massimo. Se il file è vuoto, la funzione deve restituire `None`.

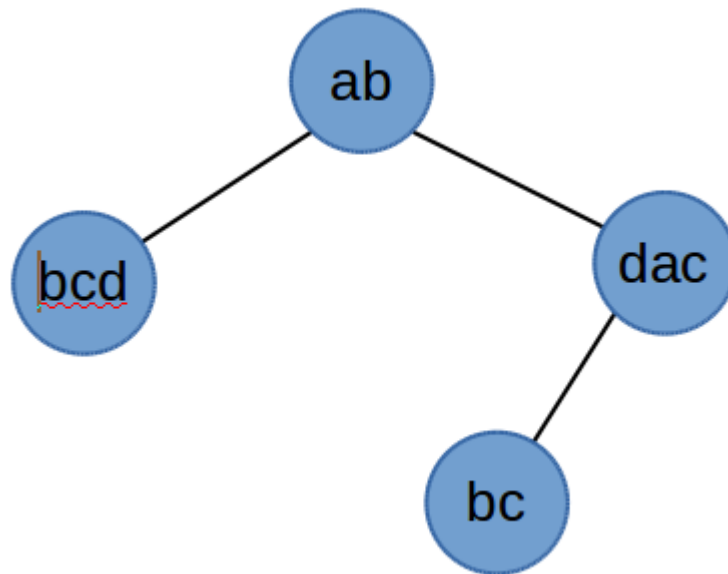
**Esempio:** se il file contenesse i dati:

*"Giulio,10,3,5\n Giovanna,6,5,6\n Marta,10,2,2"*

allora la funzione dovrebbe restituire il numero `10`.

4. **A\_Ex4(a, s) (5 punti)** Implementare la funzione `A_Ex4(a, s)` che riceve in ingresso un oggetto `a` della classe `BinaryTree` (la cui interfaccia è riportata sotto) e una stringa `s`. Il primo parametro della funzione rappresenta un albero binario, i cui nodi hanno etichette di tipo stringa. La funzione deve restituire la *lunghezza complessiva delle etichette dei nodi dell'albero che hanno la stringa `s` come prefisso*. Se l'albero è vuoto la funzione deve restituire il valore 0.

Ad esempio, se l'albero fosse quello della figura sottostante e `s` contenesse la stringa "bc" la funzione dovrebbe restituire il valore 5.



Si ricorda che, se `s1` e `s2` sono stringhe, `s1.startswith(s2)` restituisce `True` se `s2` è un prefisso di `s1`, `False` altrimenti.

La classe `BinaryTree` implementa, tra gli altri, i metodi descritti dalla seguente interfaccia:

```
class BinaryTree:
    ## Il generico oggetto di questa classe rappresenta la radice di un (sotto)albero
    ## binario e contiene i riferimenti agli eventuali figli destro e sinistro
    def __init__(self, rootObj):
        ## Inizializza un albero. rootObj è l'identificatore della radice

    def getRightChild(self):
        ## Restituisce la radice del sottoalbero sinistro o None (se il nodo corrente
        ## non ha un figlio sinistro)

    def getLeftChild(self):
        ## Restituisce la radice del sottoalbero sinistro o None (se il nodo corrente
        ## non ha un figlio sinistro)

    def getRootVal(self):
        return self.key
        ## Restituisce l'identificatore associato al nodo rappresentato da self

    ## Altri metodi
```

Si noti che potreste non aver bisogno di tutti i metodi (o del costruttore) elencati sopra. Si noti anche che copia dei dati di test a partire dai quali sono costruiti gli oggetti passati alla funzione si trovano nei file `file1.txt ... file5.txt`. Non modificare per nessun motivo gli originali contenuti nella sottocartella `files`