

# 14 Settembre 2018

---

## Esame di Fondamenti di Informatica per Ing. Gestionale

---

### Compito B - 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 `B_ExN.py` (dove N è il numero dell'esercizio) con lo scheletro della soluzione. Il file `B_S_ExN.pyc` contiene la soluzione precompilata dell'esercizio proposta dal docente. L'unico file da modificare è `B_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 eventuali errori. *Ricordiamo comunque che in sede di correzione da parte dei docenti le vostre soluzioni verranno provate su dati di test diversi da quelli usati durante l'esame*

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. **B\_Ex1(l) (9 punti)** Completare la funzione Python `B_Ex1(l)` che prende in ingresso una lista `l` di stringhe. La funzione deve restituire una lista che per ogni stringa `s` in `l` contiene una stringa `s'` ottenuta concatenando il primo e l'ultimo carattere di `s`. Se `s` è una stringa di un solo carattere, questo va (ovviamente) considerato contemporaneamente come il primo e l'ultimo carattere di `s`. Se `s` è una stringa vuota, la stringa `s'` ottenuta da `s` deve essere una stringa vuota. Le stringhe nella lista restituita *devono essere ordinate alfabeticamente*. Se la lista `l` in input è vuota, la funzione deve restituire una lista vuota.

**Esempio:** se `l= ["pippo", "mamme", "a", ""]`, la funzione dovrebbe restituire la lista `['', 'aa', 'me', 'po']`.

2. **B\_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 in ciascuna riga il primo elemento è uguale al primo elemento della corrispondente riga in `m`, mentre gli altri elementi sono ottenuti moltiplicando il corrispondente elemento di `m` per l'elemento che in `m` lo precede nella riga. 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

$$\begin{pmatrix} 1 & 2 & 3 \\ 5 & 6 & 2 \\ 1 & 3 & -8 \end{pmatrix}$$

la funzione dovrebbe restituire la matrice

$$\begin{pmatrix} 1 & 2 & 6 \\ 5 & 30 & 12 \\ 1 & 3 & -24 \end{pmatrix}$$

invece se la matrice fosse

$$\begin{pmatrix} 1 & 2 & 3 \\ 5 & 6 & 2 \end{pmatrix}$$

la funzione dovrebbe restituire `[]`

3. **B\_Ex3(file) (9 punti)** Implementare la funzione `B_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 il nome del cassiere che ha venduto, complessivamente, il numero minimo di pezzi. Se il file è vuoto, la funzione deve restituire `None`.

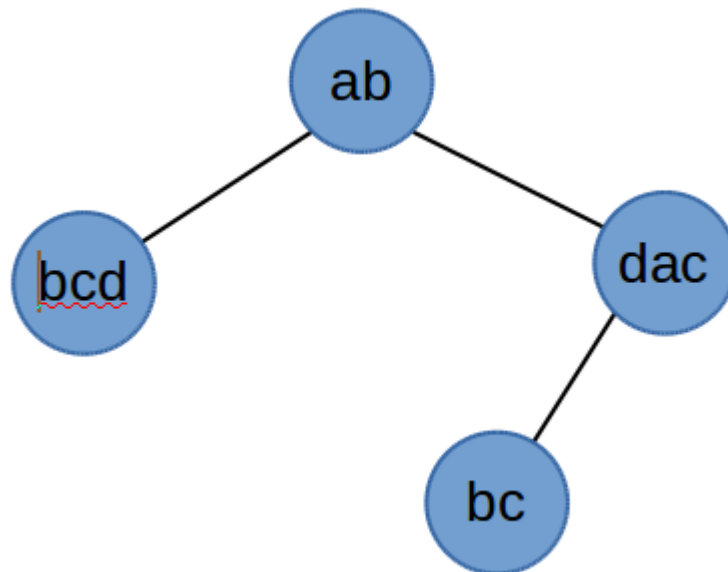
**Esempio:** se il file contenesse i dati:

*"Giulio,10,2,5\n Giovanna,6,5,9\n Flavio,2,3,5\n Giulio,3,2,4"*

allora la funzione dovrebbe restituire la stringa `'Flavio'`.

4. **B\_Ex4(a) (5 punti)** Implementare la funzione `B_Ex4(a)` che riceve in ingresso un oggetto `a` della classe `BinaryTree` (la cui interfaccia è riportata sotto). Il parametro della funzione rappresenta un albero binario, i cui nodi hanno etichette di tipo stringa. La funzione deve restituire il massimo valore della lunghezza delle etichette associate ai nodi dell'albero. Se l'albero è vuoto la funzione deve restituire il valore 0.

Ad esempio, se l'albero fosse quello della figura sottostante la funzione dovrebbe restituire il valore 3.



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 ... file8.txt`. Non modificare per nessun motivo gli originali contenuti nella sottocartella `files`