

13 Febbraio 2019

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 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. **A_Ex1(l,c) (9 punti)** Implementare la funzione Python `A_Ex1(l1,l2,n)` che prende in ingresso due liste di interi `l1` e `l2` e un intero `n`. La funzione deve restituire `True` se la somma dei primi `n` elementi di `l1` con gli ultimi `n` elementi di `l1` è uguale alla somma dei primi `n` elementi di `l2` con gli ultimi `n` elementi di `l2`. Se `n` è maggiore della lunghezza di `l1` o di `l2` la funzione deve restituire `False`.

Esempio: Se `l1=[1,2,3]`, `l2=[2,3,1,2]` ed `n=2`, la funzione deve restituire `True` perché la somma dei primi due elementi con gli ultimi due elementi sia in `l1` che in `l2` è 8.

Se invece `l1 = [7,2,3,1,4]`, `l2 = [12,2,3,4,14]` e `n = 1` la funzione deve restituire `False` in quanto la somma del primo elemento di `l1` con l'ultimo elemento di `l1` è uguale a 11, mentre la somma del primo elemento di `l2` con l'ultimo elemento di `l2` è uguale a 26.

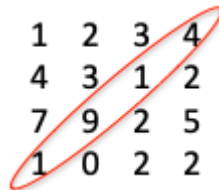
2. **A_Ex2(m) (9 punti)** Implementare la funzione Python `A_Ex2(m)` che prende in input una matrice quadrata `m` e restituisce la somma degli elementi sulla diagonale secondaria.

Esempio: Se la matrice in input è `[[0,1,4],[1,0,0],[2,4,5]]`, cioè rappresenta la matrice

$$\begin{pmatrix} 0 & 1 & 4 \\ 1 & 0 & 0 \\ 2 & 4 & 5 \end{pmatrix}$$

la funzione deve restituire 6 (4+0+2).

Nota: Potete assumere che la matrice in input sia sempre quadrata. Si ricorda inoltre che la diagonale secondaria di una matrice è composta dagli elementi che vanno dall'angolo in alto a destra a quello in basso a sinistra come mostrato nella seguente figura:



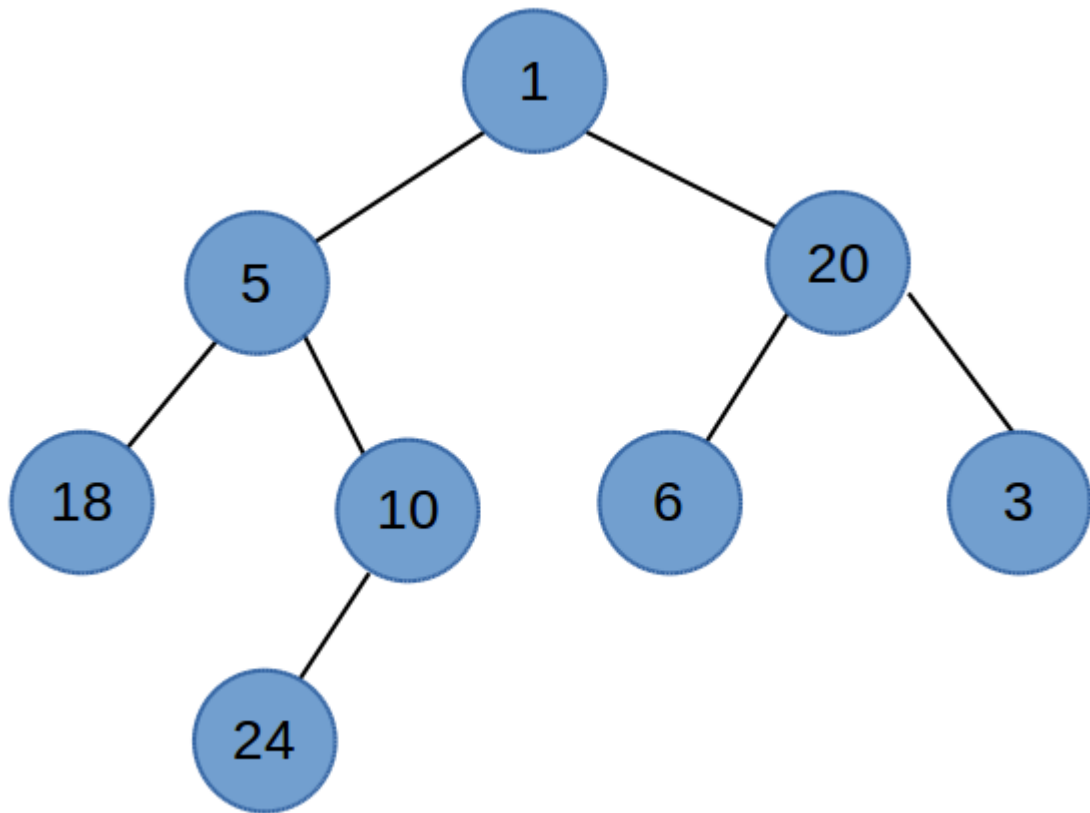
1	2	3	4
4	3	1	2
7	9	2	5
1	0	2	2

3. **A_Ex3(file) (9 punti)** Implementare la funzione Python `A_Ex3(file)` che prende in ingresso il nome di un file contenente la lista dei risultati di alcune partite di NBA nel seguente formato `'SquadraA,SquadraB,PunteggioA,PunteggioB'`. La funzione deve restituire una lista che contiene il nome della squadra con maggior numero di vittorie fra le partite riportate nel file. Se più squadre condividono lo stesso numero massimo di vittorie la lista deve contenere i loro nomi **ordinati alfabeticamente**. Se il file è vuoto la funzione deve restituire la lista vuota `[]`.

Esempio: se il file contenesse il testo: `"C,B,103,102\nB,A,112,111\nA,C,98,109\nB,D,97,95"` allora la funzione dovrebbe restituire la lista `['B','C']`, perché sia B e C hanno due vittorie, mentre A e D non hanno nessuna vittoria.

4. Implementare la funzione Python `A_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 che rappresentano valori interi. La funzione deve restituire il numero di nodi dell'albero il cui valore è strettamente maggiore della somma di quelli dei figli. Se l'albero è vuoto la funzione deve restituire il valore 0, mentre se la radice non ha figli la funzione deve restituire 1 (in pratica, le foglie soddisfano sempre la condizione).

Esempio: se l'albero fosse quello della figura sottostante la funzione dovrebbe restituire il valore 5. Infatti, l'albero ha 4 foglie. Inoltre, il nodo di etichetta 20 soddisfa la condizione.



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`.