

05.05.2017

# Esame di Fondamenti Informatica per Ingegneria Gestionale - A.A. 2016/2017

## Durata 1h45' - Compito A

### Istruzioni (leggere attentamente)

**Nota importante:** 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). Questo è il file che dovreste modificare con la vostra soluzione. Non create nuovi file.

Per verificare la correttezza di un esercizio **DOVETE** usare il programma TestEx.pyc (basta cliccarci sopra 2 volte) che proverà la vostra soluzione con un certo numero di casi di test. **Si noti che per la correzione verranno usati insieme di dati di test diversi.**

E' possibile consultare la documentazione ufficiale del linguaggio Python, ma **non è possibile usare libri o appunti**. In caso di problemi tecnici chiedere ai docenti o ai tecnici del laboratorio.

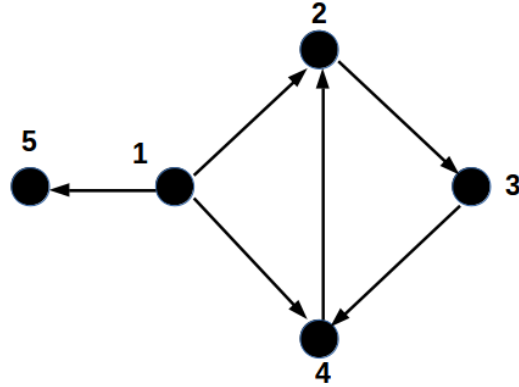
#### Esercizi

- **A\_Ex1(l,c)** Scrivere una funzione che riceve in ingresso una lista di stringhe **l** ed un carattere **c** e restituisce una lista composta dalle sole stringhe di **l** in cui **c** compare esattamente **2** volte. Ad esempio, se **l = ['faccia', 'asta', 'canasta', 'pane', 'cane']** e **c = 'a'** allora deve restituire la lista **['faccia', 'asta', 'canasta']**
- **A\_Ex2(file)** Scrivere una funzione che prende in ingresso un **file** che contiene l'altezza di **n** persone misurata ogni anno e restituisce il nome della persona che è la più alta di tutti in tutte le misurazioni. Se tale persona non esiste la funzione deve restituire **None**. Ad esempio, se le righe del file sono "Marco, Lucio, Giovanni", "1.50, 1.30, 1.45", "1.55, 1.50, 1.49" la funzione deve restituire "Marco".
- **A\_Ex3(file) (8 punti)** Completare la funzione Python che riceve in ingresso una matrice di interi **m** e restituisce l'indice della riga **i** il cui prodotto è minimo. Se la matrice è vuota allora la funzione deve restituire il valore -1. Se più di una riga ha lo stesso prodotto minimo, allora la funzione deve restituire l'indice maggiore tra quelli con prodotto minimo.

Esempio: Se la matrice **m** è  $\begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}$  allora la funzione deve restituire 0.

- **A\_Ex4(g, u, v) (8 punti)** Implementare la funzione **A\_Ex4(g, u)** che (tra gli altri parametri) riceve in ingresso un dizionario **g** che rappresenta un grafo diretto, i cui nodi hanno etichette che sono valori interi. Il parametro **u** è un nodo del grafo. La funzione deve restituire il valore medio

(arrotondato all'intero immediatamente inferiore) delle etichette associate ai nodi che sono raggiungibili da **u**. Si ricordi che un nodo *y* è raggiungibile a partire da un nodo *x* se esiste un cammino (diretto nel nostro caso) che parte da *x* e raggiunge *y*.



Ad esempio, se il grafo fosse quello in figura, avremmo  $\mathbf{g} = \{1: [4, 2, 5], 2: [3], 3:[4], 4:[2], 5: []\}$ . Se **u** fosse il nodo 4, la funzione dovrebbe restituire il valore 3, perché 3 nodi sono raggiungibili a partire dal nodo 4 (i nodi 2, 3, e 4) e la somma delle loro etichette è 9. Si consideri che il nodo di partenza è raggiungibile per definizione.