

14BHD INFORMATICA, A.A. 2020/2021

Esercitazione di Laboratorio 1

Obiettivi dell'esercitazione

- Esercitarsi con i flowchart e lo pseudocodice
- Prendere confidenza con l'ambiente di sviluppo
- Familiarizzarsi con il linguaggio Python

Contenuti tecnici

- Uso dei vari tool per la programmazione
 - Creazione ed esecuzione di script Python
 - Stampe a terminale
-

Da risolvere in laboratorio

- Esercizio 1. Volete calcolare la percentuale di utilizzo della vostra automobile per uso personale e, separatamente, per recarvi al lavoro. Conoscete la distanza tra casa vostra e il vostro luogo di lavoro e, per un certo periodo, avete registrato il valore iniziale e finale riportato dal contachilometri, oltre al numero di giorni lavorativi. Scrivete un algoritmo per risolvere questo problema. [R1.16]
- Esercizio 2. Supponete di andare in un ristorante di lusso insieme ad alcuni amici e che, al momento di pagare il conto, vogliate dividere in parti uguali, compresa la mancia del 15%. Descrivete, mediante un diagramma di flusso, un algoritmo che calcoli la somma dovuta da ciascuno. Il programma deve visualizzare l'importo del conto, la mancia, il costo totale e la somma dovuta da ciascuno; inoltre, deve visualizzare la somma dovuta da ciascuno per pagare il conto e, separatamente, per la mancia. [R1.20]
- Esercizio 3. In un programma di calendarizzazione (scheduling) di eventi si deve verificare se due appuntamenti si sovrappongono. Per semplicità, ipotizziamo che gli appuntamenti inizino sempre a un'ora esatta (senza minuti) e usiamo l'orario militare (cioè con le ore che vanno da 0 a 23). Lo pseudocodice seguente descrive un algoritmo che determina se l'appuntamento che inizia all'ora start1 e termina all'ora end1 si sovrappone all'appuntamento che inizia all'ora start2 e termina all'ora end2.

Se $start1 > start2$
 $s = start1$
Altrimenti
 $s = start2$
Se $end1 < end2$
 $e = end1$
Altrimenti
 $e = end2$
Se $s < e$
 Gli appuntamenti si sovrappongono
Altrimenti
 Gli appuntamenti non si sovrappongono

- i. Seguite passo dopo passo l'esecuzione del codice con gli appuntamenti 10–12 e 11–13, poi con gli appuntamenti 10–11 e 12–13
- ii. Disegnate il diagramma di flusso per l'algoritmo [R3.12]

Esercizio 4. L'algoritmo seguente individua la stagione (Spring, Summer, Fall o Winter, cioè, rispettivamente, primavera, estate, autunno o inverno) a cui appartiene una data, fornita come mese e giorno.

Se mese è 1, 2 o 3, stagione = "Winter"
Altrimenti se mese è 4, 5 o 6, stagione = "Spring"
Altrimenti se mese è 7, 8 o 9, stagione = "Summer"
Altrimenti se mese è 10, 11 o 12, stagione = "Fall"
Se mese è divisibile per 3 e giorno ≥ 21
 Se stagione è "Winter", stagione = "Spring"
 Altrimenti se stagione è "Spring", stagione = "Summer"
 Altrimenti se stagione è "Summer", stagione = "Fall"
 Altrimenti stagione = "Winter"

Disegnate il diagramma di flusso per l'algoritmo [R3.13]

Da risolvere a casa

Esercizio 5. Scrivete un programma che visualizzi la somma dei primi 10 numeri interi positivi: $1 + 2 + \dots + 10$. [P1.2]

- Esercizio 6. Scrivete un programma che visualizzi il saldo di un conto bancario dopo il primo, secondo e terzo anno. Il conto ha un saldo iniziale di 1000 dollari e vi vengono accreditati interessi annuali al 5%. [P1.4]
- Esercizio 7. Nel moderno mercato del lavoro, la capacità di parlare più lingue è un'abilità decisamente apprezzata ed è basilare imparare i saluti. Scrivete un programma che visualizzi un elenco, su due colonne, con frasi di saluto: nella prima colonna scrivete frasi in inglese (ad esempio, "Good morning", "It is a pleasure to meet you", "Please call me tomorrow", "Have a nice day!"), mentre nella seconda scrivetene la traduzione in italiano. [P1.16]
- Esercizio 8. Scrivete un programma che visualizzi sullo schermo il vostro nome all'interno di un rettangolo, come nell'esempio seguente:

```
+-----+  
| Dave |  
+-----+
```

Fate quanto possibile per comporre i lati del rettangolo con i caratteri | - +. [P1.5]