

## Esercitazione Python del 13 marzo 2015

Obiettivo dell'esercitazione è di prendere confidenza con Python e con il suo ambiente IDLE e di fare pratica con l'istruzione WHILE. Per risolvere gli esercizi descritti di seguito dovete usare l'editor di IDLE che si apre cliccando sull'icona di Python sul desktop e poi selezionando File -> New File. Scrivete in questo file la vostra soluzione e poi salvatela sul desktop. Usate file diversi per i vari esercizi, ad esempio chiamando i file "esercizio1.py", "esercizio2.py" e così via. Ricordatevi di copiare i file sulla vostra pennetta USB prima di andare via.

### Funzioni Utili

Per risolvere gli esercizi possono essere utili queste funzioni Python:

- **len(stringa)** restituisce la lunghezza della stringa
- **ord(carattere)** restituisce il numero corrispondente al carattere nella tabella UNICODE
- **chr(numero)** restituisce il carattere corrispondente al numero nella tabella UNICODE

L'operatore **in** di Python permette di verificare se un carattere (o una stringa) appartengono ad un'altra stringa. Ad esempio:

- "a" in "palla" avrà come risultato **True**                      "b" in "palla" avrà come risultato **False**
- "alla" in "palla" avrà come risultato **True**                      "allo" in "palla" avrà come risultato **False**

I metodi delle stringhe utili per risolvere gli esercizi sono applicati, ad esempio, a una stringa s:

- **s.replace(str1,str2,2)** crea una nuova stringa partendo da s e sostituendo le prime 2 occorrenze di str1 con str2. Notate che str1 e str2 possono essere anche stringhe di lunghezza qualsiasi e che se si vuole sostituire tutte le occorrenze allora il numero si può omettere, ad esempio s.replace("ab","bac")
- **s.find(str)** calcola la prima posizione di s in cui compare la stringa str. Se la stringa str non compare in s restituisce -1
- **s.count(str)** calcola quante volte str compare in s

### Esercizi

- 1) Scrivere un programma che prende in ingresso una sequenza di interi (per terminare inserire "\*\*\*") e restituisce il loro prodotto.
- 2) Scrivere un programma che prende in ingresso una sequenza di interi e stampa, per ogni intero, il resto della divisione per 5. Se il resto è zero, il programma termina.
- 3) Scrivere un programma che prende in una sequenza di interi e termina quando la somma di due interi adiacenti vale 10.
- 4) Scrivere un programma che prende in ingresso una sequenza di interi e termina quando la somma di due interi adiacenti è uguale all'intero successivo.
- 5) Scrivere un programma che prende in ingresso una stringa e la scandisce, carattere dopo carattere, sino a quando o la stringa finisce o la somma dei codici unicode dei caratteri scanditi è >200. In ogni caso, il programma stampa la somma calcolata e il MOTIVO per cui è terminato.
- 6) Scrivere un programma che prende in ingresso una stringa lunga almeno due caratteri e la scandisce, carattere dopo carattere, sino a quando o la stringa finisce o il codice unicode del carattere di indice i è uguale a quello del carattere precedente (di indice i-1) +10 (provare con "ak").
- 7) Scrivere un programma che prende in input una sequenza di stringhe (per terminare inserire la stringa vuota "") e restituisce l'ultima in ordine alfabetico (usate l'operatore '<' tra stringhe).
- 8) Scrivere un programma che prende in input una sequenza di stringhe e termina (stampando le ultime due stringhe) quando l'ultimo carattere della stringa precedente è uguale al primo carattere di quella attuale, ad esempio se la stringa precedente è "casa" e quella attuale è "albero" il programma deve terminare e stampare "casa" e "albero".