

Un'agenda giornaliera permette di organizzare, visualizzare e gestire gli impegni di una singola giornata. Un impegno è caratterizzato da un titolo, un orario di inizio e un orario di fine. L'agenda copre la fascia 8-24 e gli orari degli impegni devono rispettare questi limiti. Per semplicità, gli orari sono gestiti con una granularità oraria e gli intervalli sono da intendersi con estremo sinistro incluso ed estremo destro escluso. Di seguito viene riportato un esempio di agenda giornaliera con due impegni: “*fitness*” dalle ore 18 alle ore 20 ed “*esame*”, dalle ore 9 alle ore 12.

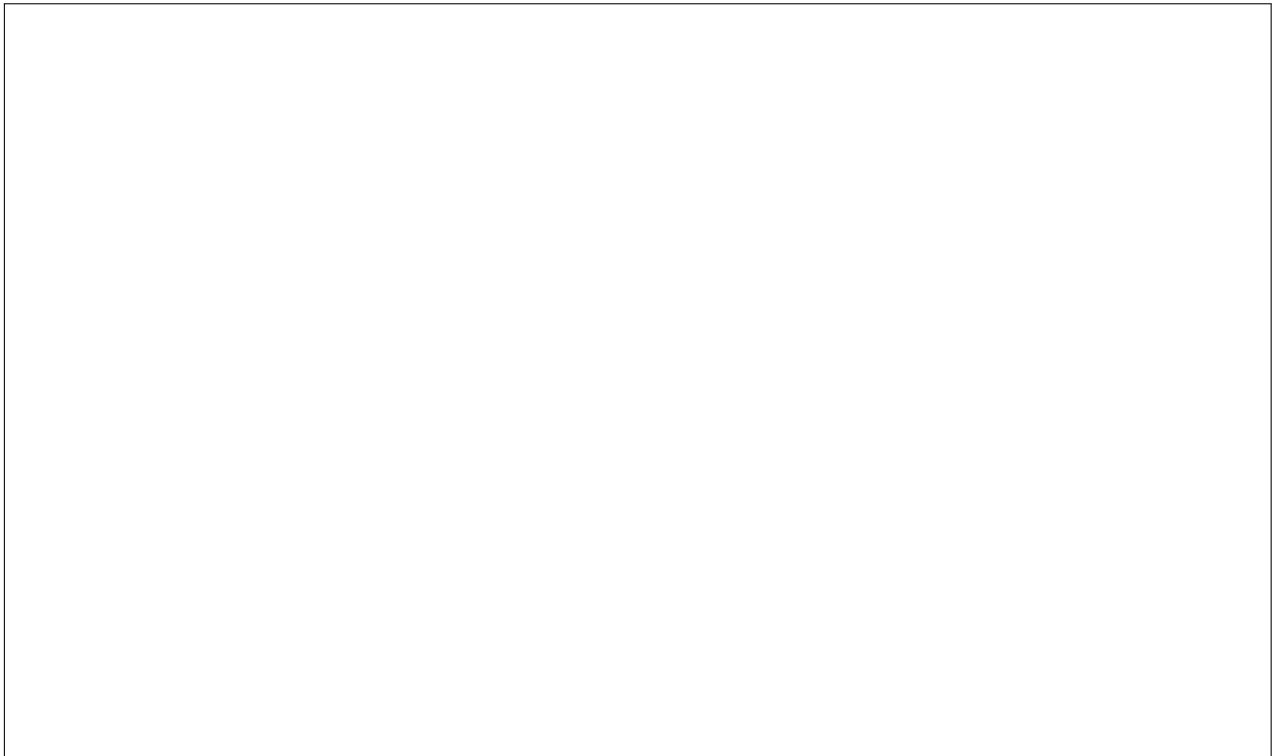
<b>fitness</b>	18-20
<b>esame</b>	9-12

L'agenda giornaliera può essere rappresentata in Python come una lista di tuple. Ogni elemento della lista corrisponde ad un singolo impegno ed è rappresentato come una tupla composta da tre elementi (titolo, orario di inizio, orario di fine).

1. Definire la funzione `ordina.impegni`

- **Parametri di ingresso:**
  - `agenda`: una lista di tuple.
- **Restituisce:** una lista di tuple.
- **Descrizione:** la funzione restituisce una lista costituita dalle tuple presenti nella lista in ingresso, ordinate in base al secondo elemento di ciascuna tupla.
- **Output atteso:** con riferimento alla seguente agenda giornaliera, [(“**fitness**”, 18, 20), (“**esame**”, 9, 12)], la lista restituita deve riportare gli impegni ordinati in base al loro orario di inizio, in ordine crescente:

[(“**esame**”, 9, 12), (“**fitness**”, 18, 20)]



2. Definire la funzione `stampa_agenda`

- **Parametri di ingresso:**
  - `agenda`: una lista di tuple.
- **Restituisce:** non restituisce alcunché.
- **Descrizione:** la funzione permette di stampare riga per riga le tuple della lista fornita in ingresso, ordinate in base al secondo elemento di ciascuna tupla, opportunamente formattate.
- **Output atteso:** con riferimento alla seguente agenda giornaliera, [(`fitness`”, 18, 20), (`esame`”, 9, 12)], la funzione stampa:

<code>esame</code>	9-12
<code>fitness</code>	18-20

- **Suggerimento:** utilizzare la funzione `ordina_impegni`.

3. Definire la funzione `inserisci_impegno`

- **Parametri di ingresso:**
  - `agenda`: lista di tuple
  - `titolo`: stringa
  - `inizio`: intero
  - `fine`: intero
- **Restituisce:** una lista di tuple.
- **Descrizione:** la funzione permette l'aggiunta condizionata di un nuovo elemento alla lista di tuple fornita in ingresso. L'elemento eventualmente aggiunto è una tupla costituita dagli elementi `titolo`, `inizio`, `fine` forniti in ingresso. La funzione inserisce un nuovo elemento solo se sono soddisfatte entrambe le seguenti condizioni:
  - l'intervallo `inizio-fine` è tale che:  $8 \leq \text{inizio} < \text{fine} \leq 24$ ;
  - l'intervallo `inizio-fine` non si sovrappone, anche parzialmente, con nessun intervallo degli elementi presenti nella lista.

In caso contrario, la funzione stampa un messaggio e restituisce l'agenda invariata.

- **Output atteso:** Con riferimento agli input `agenda = [(“fitness”, 18, 20), (“esame”, 9, 12)]`
  - `titolo=“barcola”, inizio=14, fine=18` → restituisce `[(“fitness”, 18, 20), (“esame”, 9, 12), (“barcola”, 14, 18)]`;
  - `titolo=“barcola”, inizio=14, fine=19` → stampa “Impegno incompatibile” e restituisce l'agenda invariata;
  - `titolo=“jogging”, inizio=7, fine=9` → stampa “Orari non validi” e restituisce l'agenda invariata.

4. Definire la funzione `trova_slot_massimo`

- **Parametri di ingresso:**
  - agenda: lista di tuple
- **Restituisce:** una tupla costituita da due interi (`inizio_slot`, `durata_slot`)
- **Descrizione:** la funzione determina, all'interno di un intervallo delimitato dalle costanti 8 e 24, il più ampio sottointervallo non sovrapposto a nessuno degli intervalli specificati negli elementi della lista in ingresso. La funzione restituisce inizio e ampiezza di tale sottointervallo.
- **Output atteso:**
  - agenda= [(“fitness”, 18, 20), ( “esame”, 9, 12)] → restituisce (12, 6)
  - agenda= [(“fitness”, 18, 20), ( “esame”, 9, 12), ( “barcola”, 14, 18)] → restituisce (20, 4)
  - agenda= [ ] → restituisce (8, 16)
- **Suggerimento:** utilizzare la funzione `ordina_impegni`.

