

# Esercizio 1 (1/4)

Si consideri un'azienda agricola che offre un servizio di vendita al dettaglio dei propri prodotti.

Nel periodo primaverile l'azienda si occupa della **raccolta** e della **vendita** dei seguenti prodotti:

- **fragole** (vendute a cestini)
- **asparagi** (venduti a mazzi)

Pertanto, mentre avviene la raccolta si svolge la **vendita** dei prodotti man mano raccolti, presso un **banco** dedicato.

Siano:

- **P<sub>f</sub>** il prezzo di vendita di un cestino di fragole
  - **P<sub>a</sub>** il prezzo di vendita di un mazzo di asparagi
-

# Esercizio 1 (2/4)

Al **banco** di vendita possono accedere:

- **operai agricoli**
- **acquirenti**

## **Comportamento dell'operaio agricolo.**

Si supponga che ogni operaio abbia un comportamento **ciclico**: per un numero arbitrario di volte, l'operaio **accede al banco per depositare** una quantità arbitraria di cestini di fragole e/o di mazzi di asparagi appena raccolti.

---

# Esercizio 1 (3/4)

## Comportamento dell'acquirente.

Si supponga che ogni acquirente **acceda al banco per acquistare** una quantità arbitraria di cestini di fragole o di mazzi di asparagi (per semplicità si assuma che comperi un solo tipo di prodotto, fragole oppure asparagi):

- se la quantità di prodotto richiesta **non è disponibile**, l'acquirente rinuncia all'acquisto e se ne va.
  - se la quantità di prodotto richiesto **è disponibile**, l'acquirente la preleva dal banco, paga e va via.
-

# Esercizio 1 (4/4)

Progettare un' applicazione java che regoli gli accessi al **banco**, nella quale:

- **operai e acquirenti** siano rappresentati da **thread** concorrenti,
- il **banco** sia rappresentato da un **oggetto condiviso** da tutti i thread.

Il **thread iniziale** (**main**), una volta terminati tutti gli altri thread, stampa:

- **l'incasso** totale derivato dalle vendite;
  - **l'invenduto**, ovvero quanti mazzi di asparagi e quanti cestini di fragole sono rimasti sul banco.
- Successivamente il main termina.
-

# Impostazione

## Classi da definire:

- **Banco:** è una risorsa condivisa acceduta in modo concorrente dai thread operai e acquirenti.
    - Quali variabili locali?
    - Quali metodi (necessità di sincronizzazione!)?
  - **Operaio:** thread dipendente dell'azienda
  - **Acquirente:** thread cliente dell'azienda
  - **Main:** definisce il metodo main
-

# Impostazione

