

Esercitazioni di Programmazione Sistemi Robotici con il tool PHIDIAS

Corrado Santoro

A.A. 2018-19

(Le soluzioni a tali esercizi si trovano nella directory **samples/** del repository github di PHIDIAS)

1 Esercizio 1

Si desidera gestire la restituzione delle monete di resto di un distributore automatico. I seguenti belief memorizzano la quantità di monete presenti nel serbatoio di resto:

- `fifty(N)`, numero di monete da 50 Eurocent
- `twenty(N)`, numero di monete da 20 Eurocent
- `ten(N)`, numero di monete da 10 Eurocent
- `five(N)`, numero di monete da 5 Eurocent

Implementare la procedura `change(M)`, dove M è la quantità di denaro da restituire (in centesimi), andando a determinare le varie monete da erogare, aggiornando opportunamente i belief.

Soluzioni:

- `samples/change_coins.py`
- `samples/change_coins_w_singleton.py`

2 Esercizio 2

Un sistema automatico consente il pagamento del parcheggio auto sulla base di una tariffa oraria di 1 Euro/ora. Il sistema ha una pulsantiera che permette di impostare la durata del parcheggio con tre tasti:

- **ORA**, aumenta la durata del parcheggio di un'ora
- **MEZZ'ORA**, aumenta la durata del parcheggio di mezz'ora
- **RESET**, cancella l'impostazione di durata

La pressione di uno di questi pulsanti avvia una opportuna procedura che opera di conseguenza, visualizzando la nuova durata e la tariffa richiesta.

Il sistema accetta monete da 2 Euro, 1 Euro, 50 cent, 20 cent, 10 cent; l'inserimento di una moneta comporta l'esecuzione di una procedura in cui il taglio della moneta è fornito come parametro; tale procedura tiene traccia della cifra inserita e, al raggiungimento della cifra relativa alla tariffa selezionata, emette il ticket di parcheggio.

Soluzioni:

- `samples/parking.py`

3 Esercizio 3

Si consideri un sistema per l’allocazione dei posti su un piccolo aeroplano.

I posti sono numerati da 2 a 34, e le file (solo due) sono indicate con le lettere “a” e “b”.

I posti liberi sono indicati dal belief `free (Numero, Fila)`.

Allo scopo di mantenere il bilanciamento dell’aereo, i posti vanno allocati a partire dal centro (numerazione 17) verso prua e poppa **in modo equo**.

Scrivere una procedura `allocate (N)` che permetta di allocare N posti stampandone le informazioni di numero e fila.

Esempio:

- se (ad aereo vuoto) si richiede l’allocazione di 6 posti, i posti allocati dovranno essere: 17A, 17B, 16A, 16B, 18A, 18B
- se si richiede una successiva allocazione di 4 posti, i posti allocati dovranno essere: 15A, 15B, 19A, 19B

Soluzioni:

- `samples/airplane.py`
- `samples/airplane2.py`