

Relazione esercizio verifica del sistema con modello di automa a stati finiti

Gruppo: Williams, Mandich

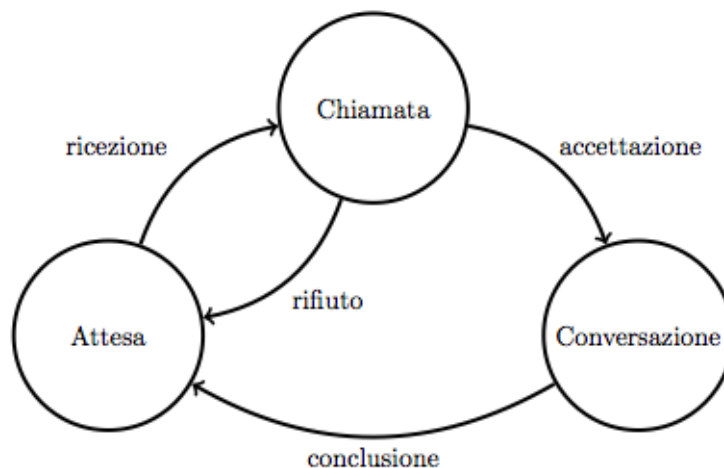
1. **Definizione problema:** si vuole progettare un ascensore tramite modello di automa a stati finiti (elaborazione del sistema tramite codice scritto in C++).
2. **Procedura:** elaborazione del codice e definire gli stati in base funzionamento dell'ascensore e dal cambio di piani (creazione del grafo per dare l'accurata descrizione del funzionamento dell'ascensore).
3. **Valutazioni e considerazioni tramite dati teorici:** per l'elaborazione dell'ascensore, è necessaria la conoscenza di un sistema ad automi a stati finiti (modello di calcolo matematico che definisce in maniera precisa e formale il comportamento dei sistemi).

Grafo: si tratta di una struttura matematica discreta che costituiscono ad un'importante parte della combinatoria (utilizzati nello studio degli automi e nelle funzioni speciali).

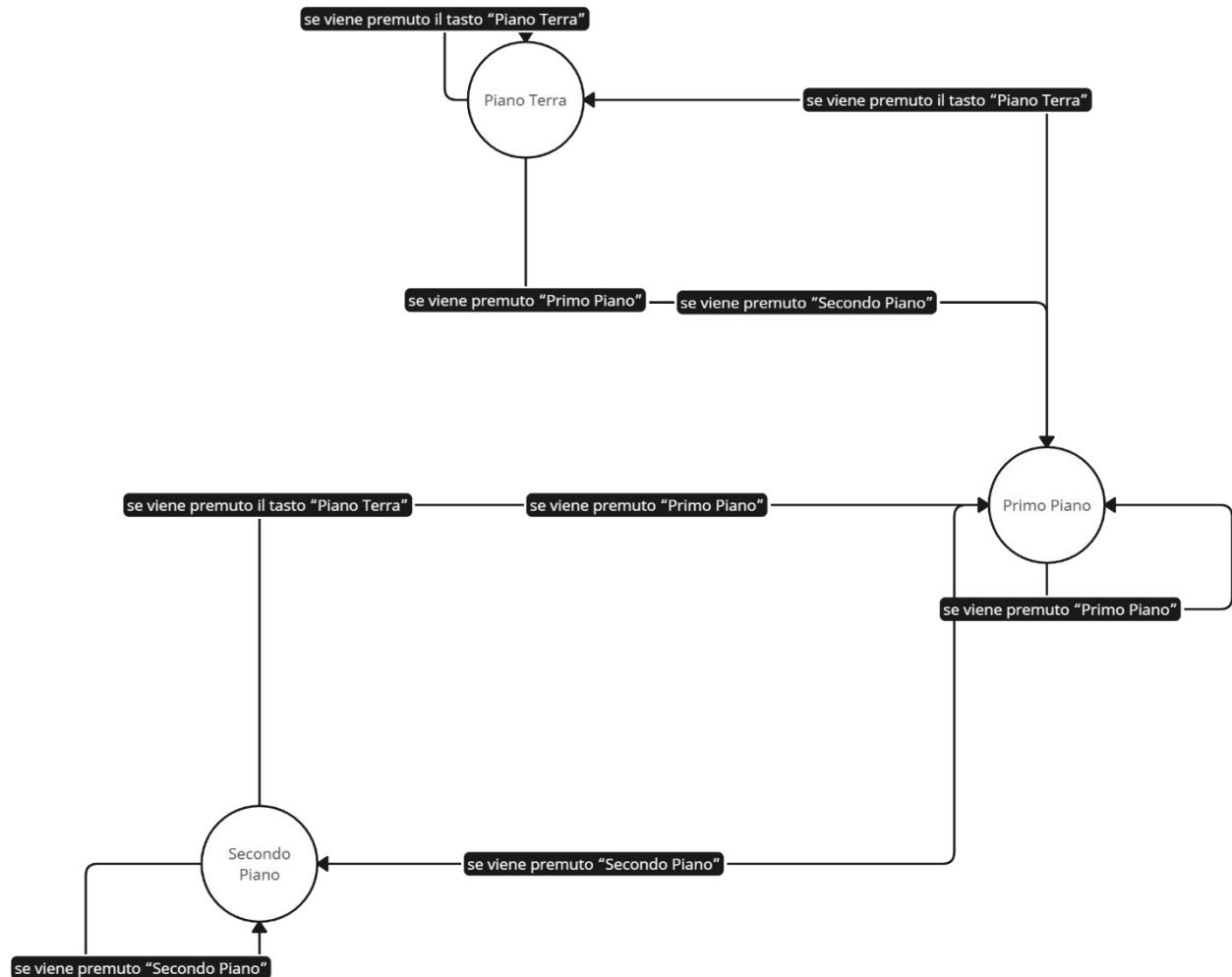
L'automa a stati finiti, per poter essere applicato, il sistema necessita di 3 caratteristiche:

- Caratteristica di evoluzione nel tempo da uno stato all'altro (Dinamicità).
- Caratteristica di espressione di valori discreti nelle variabili d'ingresso (Discretezza).
- Caratteristica di determinazione nell'uso dei simboli d'ingresso e di stati rappresentati da un numero finito (Numeri finiti).

Dal punto di vista pratico, il concetto di automa a stati finiti equivale a costruire un piccolo dispositivo che mediante una testina legge una stringa di input su un nastro e la elabora, facendo uso di un meccanismo molto semplice di calcolo e di una memoria limitata.



4. Spiegazione del codice e grafo (parte pratica):



Dichiaro una variabile booleana chiamata `running` che sarà la base dell'esecuzione del mio codice e la inizializzo a `true`

Dichiaro due variabili intere (`ascensore` e `scelta`) in cui andrò a immagazzinare lo stato della pulsantiera e del piano a cui si trova l'ascensore

Inserisco un ciclo che ha come condizione la variabile booleana precedentemente dichiarata

Se lo stato di ascensore è uguale a quello di scelta chiedo all'utente a che piano vuole andare e gli dico di scegliere il 3 se vuole terminare il programma e 0 per il piano terra

Inserisco uno switch che, in base al piano a cui voglio andare si comporta in maniera diversa

0. Se l'ascensore è al piano terra resta fermo perché è già arrivato
se l'ascensore è al primo piano va al piano terra
se l'ascensore è al secondo piano va al primo piano
1. Se l'ascensore è al piano terra va al primo piano
se l'ascensore è al primo piano resta fermo perché è già arrivato
se l'ascensore è al secondo piano va al primo piano
2. Se l'ascensore è al piano terra va al primo piano
se l'ascensore è al primo piano va al secondo piano
se l'ascensore è al secondo piano resta fermo perché è già arrivato
3. Imposto la variabile `running` a `false`

Dico all'utente a che piano si trova e a che piano deve andare (l'equivalente del led ad 8 segmenti e del pulsante luminoso)

Qui finisce il ciclo

Dico all'utente che il programma è terminato

5. Pianificazione ed organizzazione gruppo:

Mandich: stesura del codice, considerazione nella parte pratica.

Williams: preoccupazione dei dati teorici e della stesura della parte scritta.

Tempo speso: Codice: 1 ora

Grafo: 15 minuti

Raccolta dati teorici: 30 minuti.

6. Conclusioni: codice funzionante ed esito del lavoro riuscito con successo, leggere difficoltà nell'uso determinato degli elementi teorici.