

- **Librerie:**

- **robot:** libreria creata principalmente per ciò che concerne l'aspetto della mobilità del robot all'interno del labirinto. In essa troviamo infatti i corpi delle funzioni di: movimento, prossimità, rotazione oltre alla definizione di nuovi tipi di variabile.
Inoltre, qui ci occupiamo anche di qualche aspetto riguardante il labirinto come la sua dichiarazione e le funzioni che riguardano gli ostacoli al suo interno.
Per quanto riguarda la parte grafica, questa libreria si appoggia sulla libreria draw.
- **draw:** libreria creata per gestire gran parte dell'aspetto grafico del progetto: qui creiamo i vari layout dei bottoni, la matrice del labirinto, le frasi che appaiono all'interno del gioco etc...
- **game:** libreria creata a scopo di "main". Essa si occupa delle funzioni di avvio del gioco, dell'inizio della partita, del restart e del clear. Ma qui troviamo anche le varie inizializzazioni (e.g. del TIMER0, del RIT, dell'LCD e del touch, del robot etc...).

- **Joystick:**

- Il joystick ho deciso di gestirlo facendo polling attraverso il RIT. Quest'ultimo, all'inizio della partita (i.e. dopo il tocco sul labirinto), viene settato a 50ms e azionato.
- Quando una partita si conclude, il RIT viene disabilitato fino al RESET o RESTART.
- Per quanto riguarda le pressioni contemporanee di più tasti direzionali del Joystick ho optato per far sì che esse non abbiano effetto. Si entra nell'handler di tali tasti, all'interno del RIT, solo se la partita non è stata vinta e solamente se uno e uno solo dei pin dei tasti direzionali è attivo.

- **Explore:**

- Per implementare la modalità di explore ho riadattato la funzione `robot_set_dir()` del progetto `extrapoints1`.
- Nel RIT, all'interno degli handler dei tasti direzionali del joystick, una volta che uno di questi viene premuto per un secondo (o multipli), tramite la variabile globale `MOVE` (pilotata principalmente durante il gioco dal tasto `SELECT`) si riesce a capire in che modalità ci troviamo e, appunto, se siamo in `EXPLORE` viene richiamata, se necessario, la `robot_set_dir()` che si occuperà di far ruotare il robot in una delle quattro direzioni, fare il clear della cella e disegnare il robot in modo che indichi la giusta direzione.

- **Move:**

- Quando uno dei tasti direzionali del joystick viene premuto per un secondo in modalità `MOVE` (i.e. quando la variabile `MOVE` citata per l'explore vale 1) se si parte da una direzione diversa rispetto quella cui ci si vuole spostare allora assumo che il robot possa muoversi lungo quella direzione e avvio il tentativo di movimento post rotazione.
Se il robot vuol continuare a procedere verso una direzione lungo la quale non ha ancora incontrato un ostacolo, lo lascio procedere lungo essa.
Se il robot è fermo e non cambia direzione, la funzione di movimento non viene chiamata.
- Il movimento viene effettuato/provato tramite la chiamata alla funzione `step()`. Essa, in base alla direzione puntata dal robot, muove un passo in avanti e in base alla casella su cui finisce agisce di conseguenza distinguendo i casi in cui il valore della casella è 0,1,2.

- **Prossimità:**

- La `step()` che la `robot_set_dir()` utilizzano la funzione `proximity()` per implementare il sensore di prossimità che è attivo durante movimenti e rotazioni per l'appunto. Tale funzione, riadattata dal vecchio progetto, si occupa del flaggare gli ostacoli trovati entro il range di 5 spazi e richiamando la `obstacleDetected()` passandole le coordinate dell'ostacolo che, qualora l'ostacolo non fosse già stato scoperto si occupa anche di disegnarlo.

- **Ostacoli:**

- Dato che il numero di ostacoli è già noto in partenza, all'interno al primo avvio del gioco creo un vettore di tipo `_obstacle` dimensionato del numero di ostacoli. Ciascun elemento di questo vettore rappresenta un ostacolo identificato spazialmente da una variabile di tipo `_coord` e da un `char` che svolge la funzione di flag (rilevato/non rilevato).

- **Touch:**

- Per la gestione dei tocchi sul touchscreen ho deciso di utilizzare il **TIMER0** inizializzato ad un valore pari a 25ms. Mi è sembrato ragionevole imporre che i tocchi nelle zone richieste dal gioco vengano rilevati solo se della durata totale di 0.125s.
- Il rilevamento del tocco sul labirinto per far iniziare la partita viene disattivato dopo l'inizio della partita.
- Il polling per il tocco del clear e del reset resta attivo dopo l'inizio della partita e durante la partita stessa. Però quello sul clear viene disabilitato alla vittoria e l'unico tocco rilevabile sarà quello su restart.