

# **Informatica**

## **e Tecnologie della Comunicazione Digitale**

*Docente:*

**Miguel Ceriani ([ceriani@di.uniroma1.it](mailto:ceriani@di.uniroma1.it))**

*Lezioni:*

**Mercoledì/Giovedì/Venerdì 9-11**

*Ricevimento (su appuntamento):*

**Mercoledì 14-16 a viale Regina Elena 295, palazzina F, 1° piano**

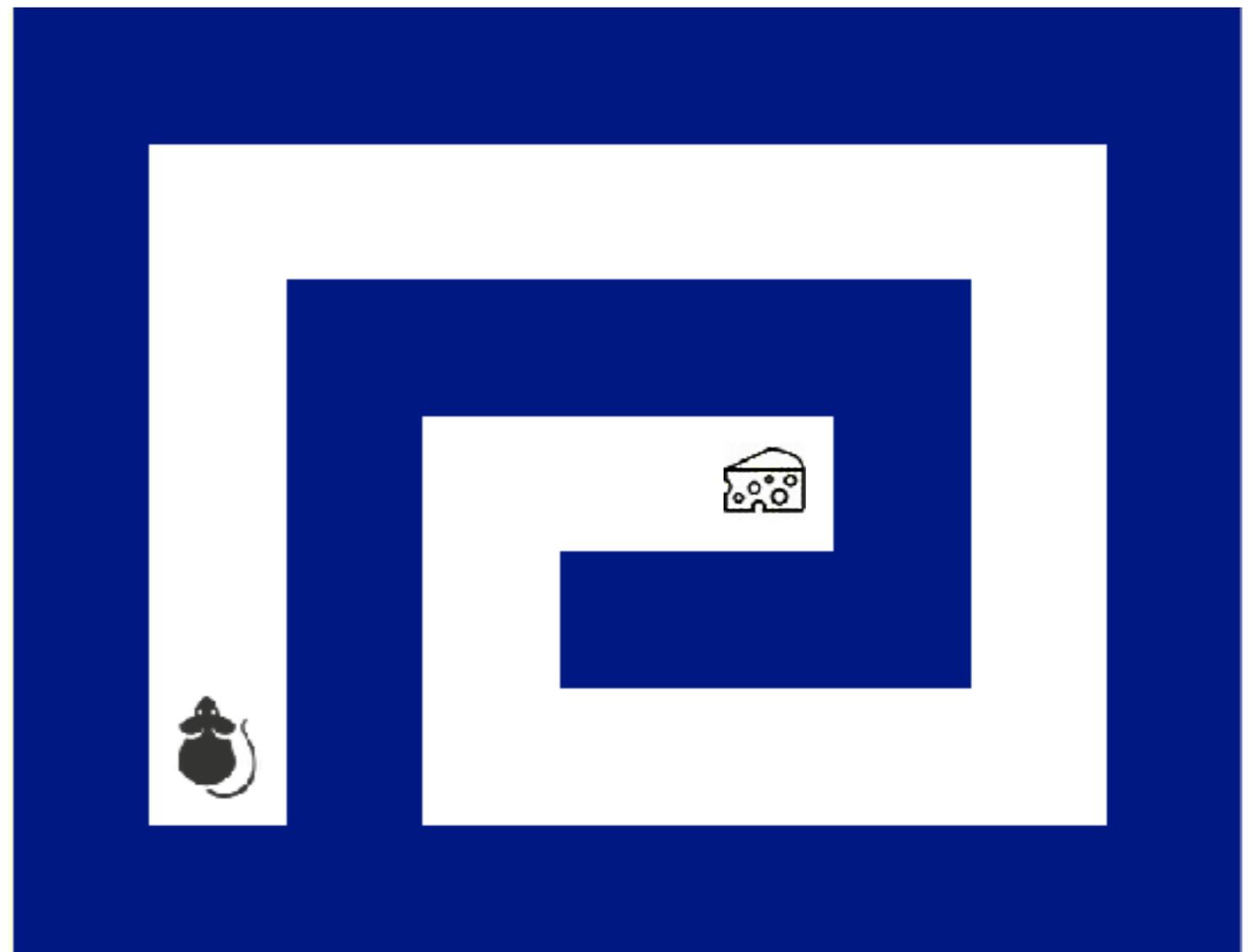
Lezione 17:  
Linguaggi di  
Programmazione:  
Soluzioni Generali

# Categorie di Labirinti

- Abbiamo risolto categorie di problemi (labirinti) di dimensione variabile, ma ancora molto specifiche: tutto dritto e una sola svolta
- Tentiamo di generalizzare, ovvero trovare programmi che risolvano categorie più ampie di casi
- L'obiettivo ideale sarebbe scrivere un programma che trovi l'uscita per qualunque labirinto (problema generale)

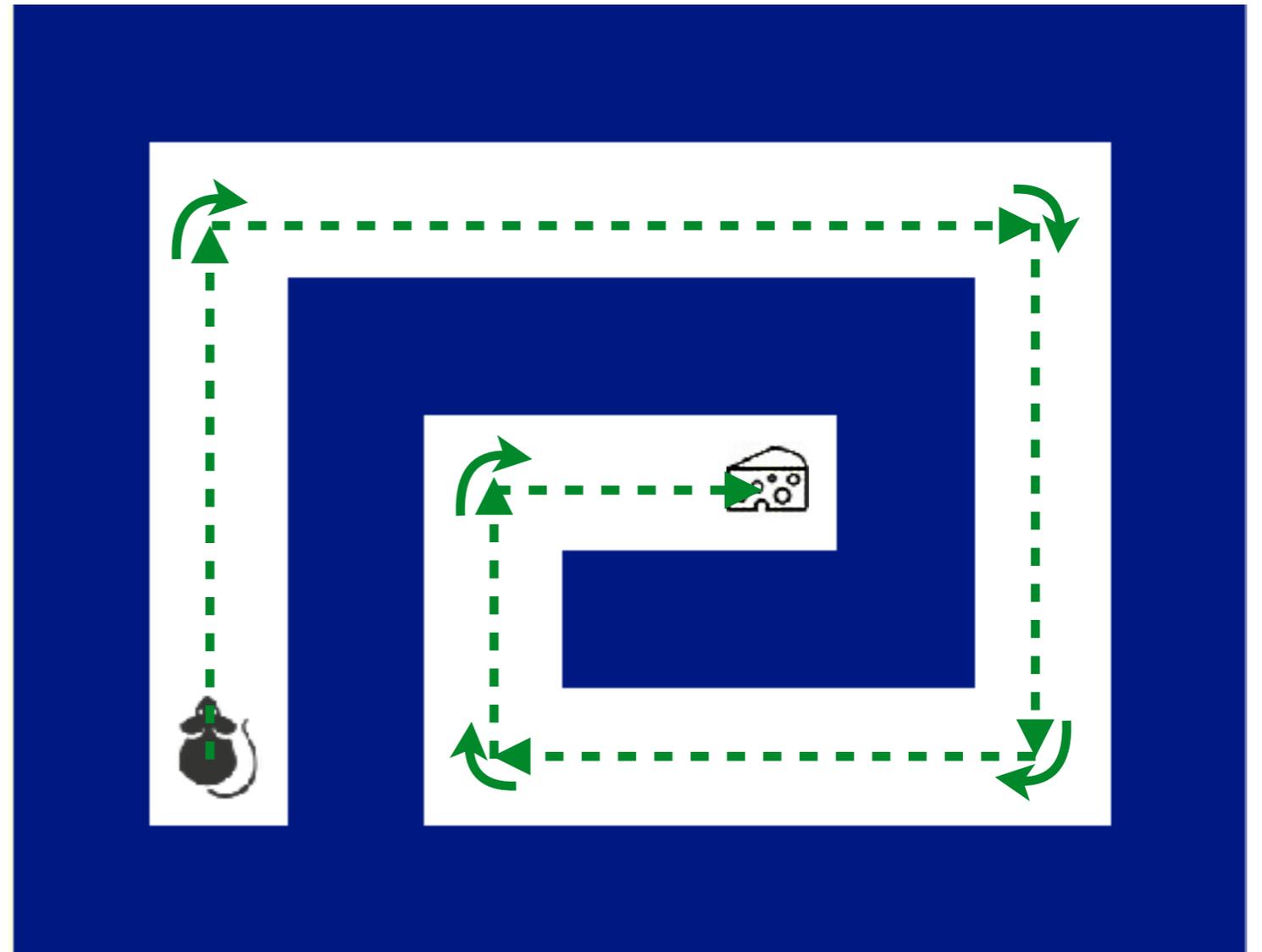
# Svolte a Destra

- qualunque lunghezza
- qualunque numero di svolte, ma sempre a destra
- nessun bivio



# Programma

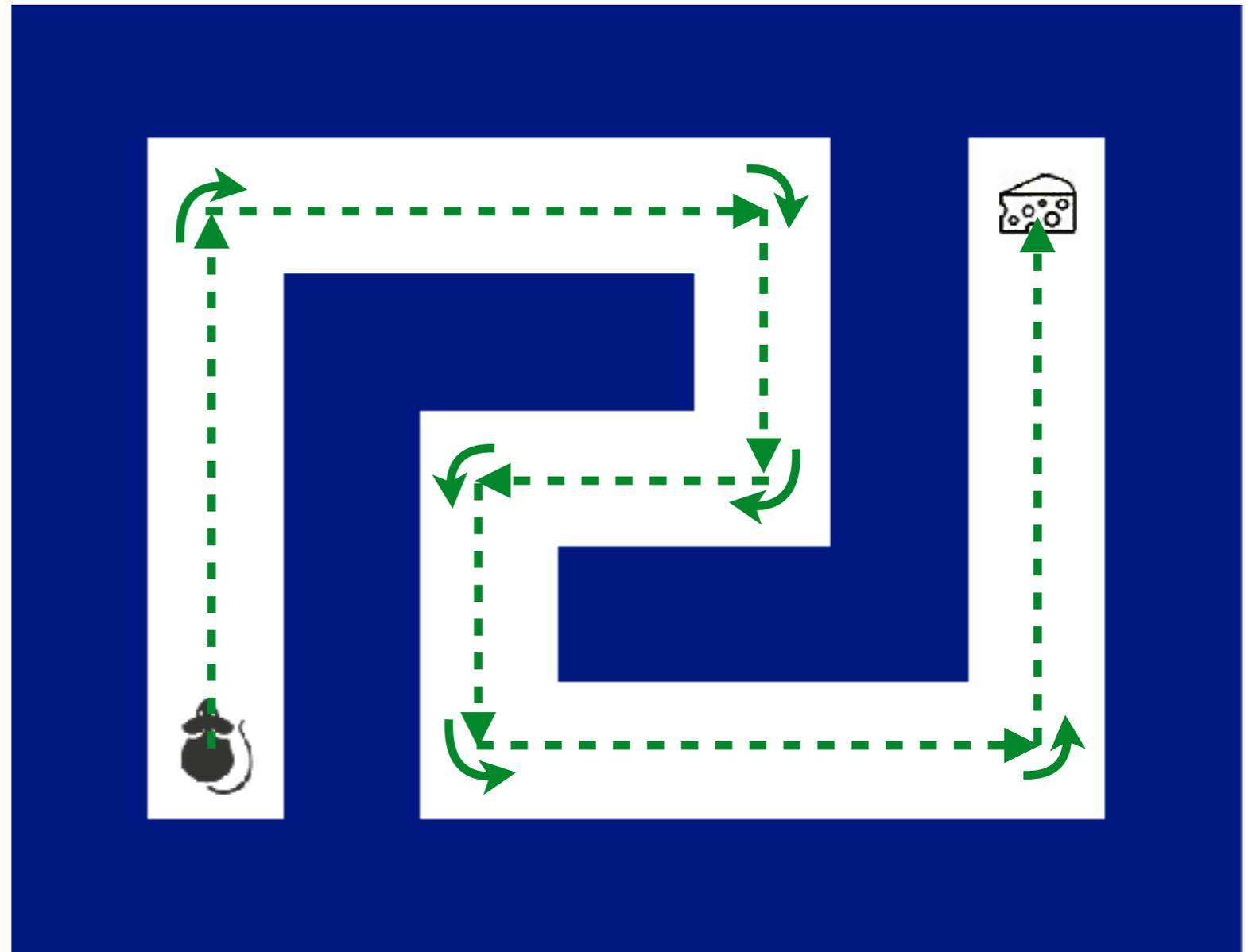
```
WHILE (NOT qui_formaggio()) {  
  IF (strada_avanti())  
  THEN {  
    avanti()  
  }  
  ELSE {  
    destra()  
  }  
}
```





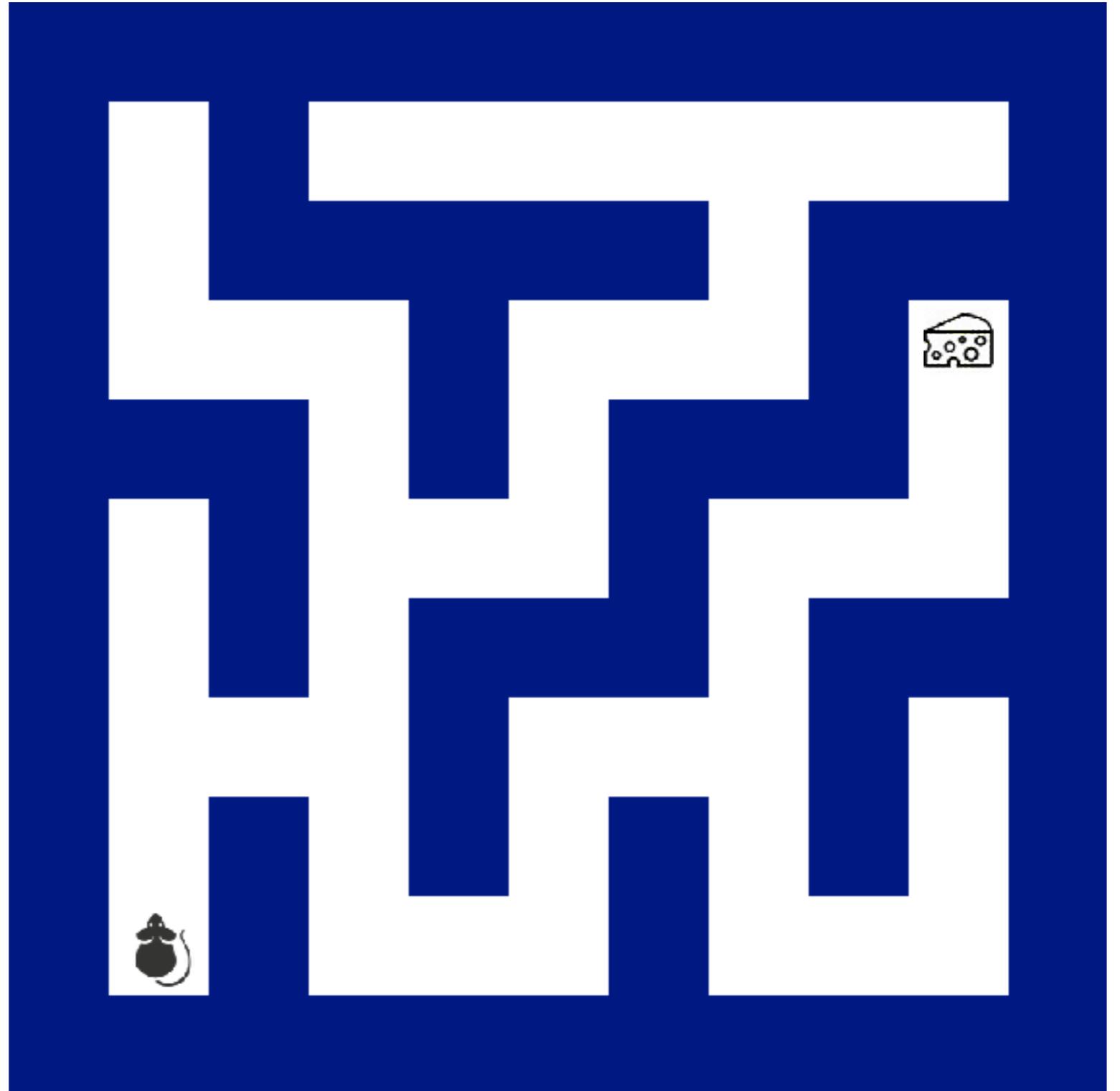
# Programma

```
WHILE (NOT qui_formaggio()) {  
  IF (strada_avanti())  
    THEN {  
      avanti()  
    }  
  ELSE {  
    IF (strada_destra())  
      THEN {  
        destra()  
      }  
    ELSE {  
      sinistra()  
    }  
  }  
}
```



# Ramificazioni senza “Circuiti”

- qualunque dimensione
- qualunque numero di svolte e ramificazioni
- nessun “circuito”: strada che riporta a un punto già attraversato





# Seguendo il Muro

- una soluzione è “seguire sempre il muro di destra” (funzionerebbe anche “seguire sempre il muro di sinistra”)
- se non ci sono circuiti, seguendo sempre il muro di destra posso percorrere tutto il labirinto, quindi sicuramente prima o poi passo dove c'è il formaggio

