

# Istruzioni programma intelligenza condivisa

Camilla Lazzati

April 2023

## 1 Compilare ed eseguire

### 1.1 Compilazione

Il progetto può essere compilato tramite il software di build automation CMake eseguendo, all'interno della directory principale, il comando

```
cmake -DCMAKE_BUILD_TYPE=Release -B build -S ., che genera i file di
configurazione necessari nella cartella ./build, seguito da cmake --build build
--config Release --target all che avvia la compilazione e salva l' eseguibile
boids in ./build.
```

### 1.2 Esecuzione

Una volta terminata la compilazione, per eseguire il programma è sufficiente eseguire `./build/boids` seguiti da eventuali parametri, la cui descrizione può essere stampata a schermo passando l'argomento `-h`.

## 2 Simulare

### 2.1 descrizione e nomi parametri

Il programma consente di modificare i parametri della simulazione da command line, quando si lancia il programma.

In particolare, i parametri da modificare per le simulazioni sono:

- tecnica d'attacco del predatore: `attack nearest (--seek-type=0)`, `attack most isolated (--seek-type=1)`, `attack center of mass (--seek-type=2)`.
- comportamento sociale/individuale: `separation factor (-s)` rappresenta la repulsione tra boid vicini, `alignment factor (-a)`, determina l'allineamento delle velocità dei boid, `cohesion factor (-c)` spinge i boid verso il centro di massa dello stormo.
- numero di predatori (`-P`) e numero di boids (`-b`). I valori di default sono rispettivamente 1 e 120.

Il programma, una volta lanciato, fa 50 simulazioni. Al termine, salva in un file i 50 valori che rappresentano il numero di vittime per ogni simulazione. Infine, stampa a schermo un riassunto dei parametri usati nella simulazione. Si consiglia di rinominare il file subito dopo il salvataggio e di segnarsi i valori dei parametri.

## 2.2 Simulazioni da fare

I possibili stati sono sei (preda sociale/individuale, ciascuna abbinata alle tre tecniche d'attacco). I valori da usare (taratura fatta sull'interfaccia grafica) sono:

- comportamento sociale: `-a 0.2, -c 0.005, -s 4.7`
- comportamento individuale: `-a 0.00001, -c 0.000015, -s 5.4`
- predatori: `-P 3`

Dunque, se voglio ad esempio lanciare la simulazione preda individuale con attack nearest, scriverò: `./build/boids -a 0.00001 -c 0.000015 -s 5.4 --seek-type=0 -P 3`

## 3 Risultati attesi

Ci si aspetta che il valor medio di vittime per comportamento sociale sia minore di quello individuale. Se così non fosse, provare a aumentare, SOLO per preda sociale, i valori hard coded in `parameters.hpp` chiamati `d_s_pred_` e `s_pred_`. Tale aumento ha senso perchè è ragionevole che l'intero stormo si accorga prima del predatore rispetto al singolo e che abbia maggiore capacità repulsiva.