

# Relazione progetto Nim

Per il seguente progetto, realizzato mediante il linguaggio C, si doveva implementare una versione del gioco Nim che si basasse su una connessione client/server locale.

Il nostro gruppo ha implementato il gioco nel seguente modo:

- `nimserver`, gestisce la connessione dei client, una volta connessi entrambi i player, la partita verrà generata tramite la creazione del campo da gioco costituito dalle 2 pile e un numero di casuale di pedine per ciascuna.
- `nimclient` permette al client di collegarsi alla lista d'attesa per una partita di `nimserver`.
- `client` si occupa gestisce le funzioni del client, ovvero la connessione al server, la ricezione di messaggi da parte del server, del controllo del vincitore ad ogni turno e invio degli input.
- `server` si occupa delle funzioni del server, ovvero l'attesa della connessione dei client, la creazione del campo da gioco, l'inizio della partita, l'invio di messaggi al client, la gestione del turno, la gestione degli input del client e della terminazione della partita in caso della ricezione di un vincitore.
- `tools` è una raccolta di varie funzioni per la gestione dei socket e dei risultati sia per il client che per il server.
- `game` per le funzioni riguardanti la partita, cioè la creazione di un campo da gioco, la scelta di una pila e del numero di pedine da rimuovere, il cambio turno la decretazione del vincitore.

Il progetto è stato testato con successo sulle seguenti macchine:

**1)**

Sistema operativo: MANJARO Linux kde 18.1.5

Versione kernel: Linux 5.4.13 x86\_64

Versione compilatore: clang 9.0.1

**2)**

(Virtual Machine)

Sistema operativo: Elementary OS 5.1 Hera (basato su Ubuntu)

Versione kernel: Linux 4.15.0 x86\_64

Versione compilatore: clang 6.0.0

Dai test effettuati è risultato che alla chiusura del thread della partita in corso lato server, dalla parte del client non viene segnalato alcun errore, semplicemente la funzione `recv` ritorna il valore 0;

Il funzionamento è riassunto nel seguente diagramma di flusso:

