

# Laboration 4 – C, C++ spel

---

Din uppgift är att implementera ett spel (klient och server) som stöder att flera klienter kan vara samtidigt uppkopplade mot servern (via TCP).

Nätverkskommunikationen i spelet skall följa specifikationen i filen SpelProtokoll.pdf.

Hantering av direction, object form, object description och chat är inget krav.

En brist i specifikationen är att den inte anger om Big eller Little endian används i kommunikationen. Om du behöver veta detta måste du ta reda på det själv.

Denna laboration ska fungera mellan flera labbgrupper. Troligen kommer ni vid redovisningen att få köra mot någon annans implementation.

## Server

Server skall implementeras i C eller C++, kompileras och köras på din linux användare (kör ssh till "ssh.student.ltu.se", använd putty eller liknande) logga in med din LTU.användare. Du får inte använda bibliotek där sockethantering är inbäddad i andra funktioner. Se [http://en.wikipedia.org/wiki/Berkeley\\_sockets](http://en.wikipedia.org/wiki/Berkeley_sockets) för godkända funktioner.

## Klient

Klienten skall implementeras i C++. Klienten kompileras och körs från din dator. Uppritandet av spelplanen skall göras av ditt GUI från laboration 3. Din Klient måste alltså ha ditt protokoll från Laboration 3 implementerat och kommunicera med ditt GUI över "localhost".

## Del A

Till din hjälp har du en färdig server som du kan/bör testa din klient mot under implementationen.

### Kommandorads klient.

Implementera en enkel kommandorads klient för att fokusera på socket programmeringen och hantering av spelprotokollet.

## Del B

Fortsätt att använda den färdiga servern.

## Klient

Implementera din klient. Kontroll av spelaren kan ske via tangentbordet eller via musen. Till servern skickas ett önskemål om att flytta sin spelare. Det är servern som bestämmer om det är tillåtet att flytta till den önskade positionen. Klienten måste hålla reda på alla spelares positioner och kommunicera detta till ditt GUI. Dessutom måste den samtidigt kommunicera med servern.

## Del C

Använd din nu fungerande klient för att testa din server.

## Server

Servern ska hantera flera samtidigt anslutna klienter. Serverns uppgift är att ta emot ett meddelande från en klient, processa innehållet och sedan ev. skicka ut uppdateringar till alla anslutna klienter. Servern ska innehålla någon form av kollisiondetektering, du får själv välja vad som ska ske vid t.ex. kollision med annan spelare eller spelplanens gränser. Det är dock inte tillåtet att befinna sig utanför spelplanen eller på samma position som annan spelare.

Se vidare information i SpelProtokoll.pdf

## Redovisning

Redovisa enligt schema. Dokumentera enligt anvisning.