

## Zadanie 2: Komunikácia s využitím UDP protokolu

Počítačové a komunikačné siete

Ema Richnáková

Piatok 08:00

FIIT STU

4. novembra 2020

# 1 Úloha zadania

Navrhните a implementujte program s použitím vlastného protokolu nad protokolom UDP (User Datagram Protocol) transportnej vrstvy sieťového modelu TCP/IP. Program umožní komunikáciu dvoch účastníkov v lokálnej sieti Ethernet, teda prenos textových správ a ľubovoľného súboru medzi počítačmi (uzlami).

Program bude pozostávať z dvoch častí – vysielacej a prijímacej. Vysielací uzol pošle súbor inému uzlu v sieti. Predpokladá sa, že v sieti dochádza k stratám dát. Ak je posielaný súbor väčší, ako používateľom definovaná max. veľkosť fragmentu, vysielajúca strana rozloží súbor na menšie časti – fragmenty, ktoré pošle samostatne. Maximálnu veľkosť fragmentu musí mať používateľ možnosť nastaviť takú, aby neboli znova fragmentované na linkovej vrstve.

Ak je súbor poslaný ako postupnosť fragmentov, cieľový uzol vypíše správu o prijatí fragmentu s jeho poradím a či bol prenesený bez chýb. Po prijatí celého súboru na cieľovom uzle tento zobrazí správu o jeho prijatí a absolútnu cestu, kam bol prijatý súbor uložený. Program musí obsahovať kontrolu chýb pri komunikácii a znovuvyžiadanie chybných fragmentov, vrátane pozitívneho aj negatívneho potvrdenia. Po prenesení prvého súboru pri nečinnosti komunikátor automaticky odošle paket pre udržanie spojenia každých 10-60s pokiaľ používateľ neukončí spojenie. Odporúčame riešiť cez vlastne definované signalizačné správy.

## 2 Analýza protokolov UDP a TCP

### Spoločné vlastnosti

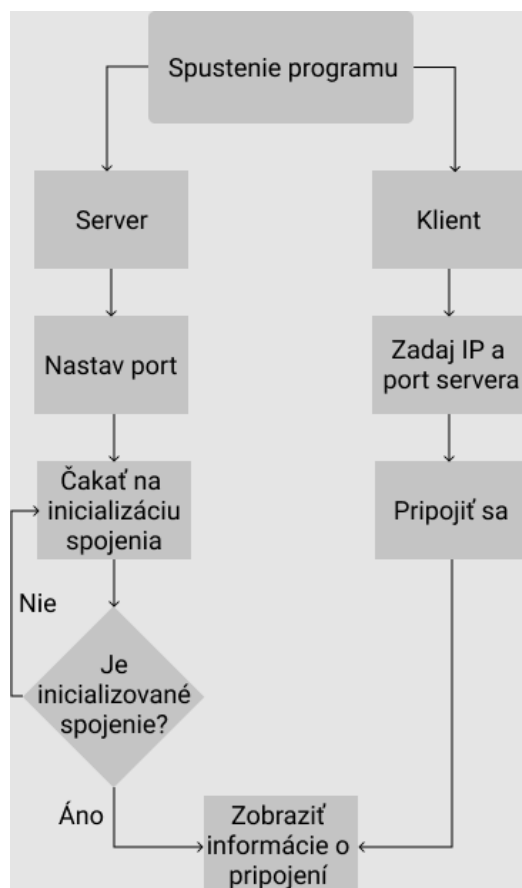
Oba protokoly sú určené na prenos dát po internete, ktoré rozdeľujú do tzv. paketov (balíčkov). Oba pracujú nad *Internet protokolom* (ozn. IP).

### Rozdielne vlastnosti

Hlavný rozdiel medzi nimi je ich účel využitia.

	TCP	UDP
Spôľahlivosť	Spôľahlivý protokol, ktorý garantuje spoľahlivosť dátového prenosu vďaka dôkladnejšej kontrole rozkúskovaných dát pri komunikácii medzi zariadeniami.	Nespoľahlivý protokol, ktorý uprednostňuje rýchlosť prenosu, a teda jeho kontrolné algoritmy sú jednoduchšie, čo môže spôsobiť neúplnosť prenosených údajov.
Kontrola chýb	Rozšírený mechanizmus kontroly. Ponúka tzv. flow control (kontrola toku dát).	Základná kontrola (checksum).
Poradie paketov	Sekvencia dát sa zachová.	Sekvencia dát sa nemusí zachovať.
Rýchlosť	Pomalší protokol, kvôli zabezpečeniu spoľahlivosti dátového prenosu cez nespoľahlivé siete.	Rýchlejší protokol, pre malý objem režijných dát a jednoduchšej kontrole pri prenose.
Strata paketov	Ak takáto situácia nastane, stratený paket zaradí na jeho správne miesto ešte pri prenose.	Stratený paket nenájde. Prijímacie zariadenie si ho musí znovu vypýtať.
Hlavička (max. veľkosť)	20-80 B	8 B

### 3 Návrh programu



Obr. 1: Inicializácia spojenia

### 4 Interakcia používateľa s projektom

Používateľovi sa po načítaní aplikácie ukáže menu, v ktorom vie vybrať možnosť *Spustiť hru*. Po spustení sa scéna presunie na bojisko. Konkrétne sa bude nachádzať na hradnom opevnení. Pohľad bude z pvej osoby, čiže bude vedieť manipulovať s kamerou. Taktiež sa na opevnení bude nachádzať aspoň jedna zbraň, ku ktorej sa používateľ vie premiestniť a vie ju tlačidlom použiť a

strielať z nej. Pri streľbe sa na obrazovke v strede zobrazí mieridlo, ktoré pomôže používateľovi lepšie mieriť na nepriateľa.

## 5 Pseudokódy

Použitie zbrane používateľom:

```
Ak hráč kolide s objektom zbrane
a ak stlačí tlačidlo Medzerník
prepne sa kamera na pohľad zbrane - začne ju ovládať
ľavým tlačidlom myši používateľ strieľa a myšou ovláda kameru
znovu stlačenie medzerníku používateľ zruší používanie zbrane
```

Útok nepriateľa na hrad:

```
Ak nepriateľ pretne hranu obrazovky
zniži sa počet životov hradu o toľko
koľko poškodenia dáva nepriateľ
```

Zničenie nepriateľa:

```
Ak projektil kolide s nepriateľom
a je to slabý nepriateľ
nepriateľa zničí
inak ak je to silný nepriateľ
uberie mu polovicu života
```

## 6 Metódy renderovania štruktúr na obrazovku

Renderovanie bude prebiehať v reálnom čase, keďže scéna bude interaktívna, prvky budú manipulovateľné a dynamické (napríklad nepriatelia, ktorí po zničení zmiznú). Na osvetlenie/reflekcii svetla sa bude používať Phongov algoritmus. Na tieňovanie bude použitý bump mapping.

## 7 Zapojenie jednotlivých komponentov aplikácie do funkčného celku

Obloha tvorí pozadie scény a zároveň vytvára obmedzený priestor, kde sa môžu nepriateľské jednotky pohybovať. V strede tejto scény sa nachádza

hrad na vznášajúcom sa ostrove. Okolo neho poletujú určenými smermi nepriateľské letecké jednotky. Používateľ sa nachádza na hradnom opevnení spolu so zbraňami. Používateľ vidí svet z prvej osoby, čiže priamo ovplyvňuje sklon, smer a pohyb kamery. Používateľ sa vie pohybovať v obmedzenom priestore hradného opevnenia. Vie interagovať so zbraňami na opevnení tak, že preberie nad nimi kontrolu a vie nimi odstrelovať nepriateľov. Projektil zbrane po kolízii s nepriateľom uberie nepriateľovi životy. Používateľ taktiež vie odísť od zbrane a znovu sa vie pohybovať po opevnení. Hlavné osvetlenie bude tvoriť slnko. Druhý zdroj osvetlenia budú fakle pripevnené na vnútorné steny opevnenia. Oheň by mal taktiež reagovať na vietor a teda mal by sa podľa jeho smeru nakláňať. Vietor by taktiež mohol ovplyvňovať trajektóriu projektilu zbrane.