

Kursnamn: Nätverksprogrammering	Kurskod: D0036D
Antal timmar: 5	Datum: 20190826

Tillåtna hjälpmedel: Inga

Jourhavande lärare m fullständigt telefonnr: Örjan Tjernström 0910-585716	Jourhavande lärare m fullständigt telefonnr:
Jourhavande lärare m fullständigt telefonnr:	Jourhavande lärare m fullständigt telefonnr:

Betygsgränser: 50% betyg 3. Gränser för betyg 4 och 5 bestämmes senare.	Antal uppgifter och totalpoäng: 7 40
---	---

Övriga uppgifter: Elektronisk utrustning skall vara avstängd under tentamen. Svaren på nedanstående uppgifter skall vara relaterade till aktuell kurs för att kunna ge poäng.
--

Allmänna anvisningar

Kontrollera att du fått samtliga uppgifter.
Besvara endast en uppgift per lösningsblad. Skriv inte på baksidan.
Skriv tydligt, texta gärna och använd inte röd penna.

Efter tentamen

Tentamensresultat meddelas senast tre veckor efter tentamenstillfället och senast två veckor före nästa omtentamenstillfälle.
Tentamensresultatet syns på Mitt LTU – Ladok för studenter.
Din rättade tentamen skannas och blir synlig på Mitt LTU – Rättade tentor.

Uppgifter till tryckeriet

Projektnummer: 341980	Antal exemplar: 3	Antal sidor: 3 inkl försättsblad
Övriga upplysningar:		

Dina svar skall endast gå att tolka på ett sätt.

1. **Molntjänster:**

- a. Ni har fått ett uppdrag att utveckla en nätverksbaserad tjänst som för varje användare tar emot en fil, krypterar den och skickar en krypterad version tillbaka till användaren. Beskriv en arkitektur av tjänsten som är anpassat för att skaleras/skalas ut. (2p)
- b. Beskriv kortfattat funktionalitet av en lastbalanserare. (2p)

2. **Nätverk och spel:(3p)**

- a. Vad är dead reckoning i sammanhang med spel och nätverk?
- b. Vilken typ av information/data skulle man använda för att implementera dead reckoning i ett biltävlingsspel?
- c. Vilket problem kan uppstå när man använder sig av dead reckoning och beskriv två metoder för att lösa det?

3. **Nätverk och spel:** För att implementera ett multiplayer spel över nätverk finns olika strategier.(4p)

- d. Beskriva en möjligt implementation av en client-server arkitektur och av en peer to peer arkitektur.
- e. Vilka för och nackdelar har dessa två?

2. **URL:**

- a. Vilka delar består en URL av? (1p)

3. **Non-Blocking IO** är ett sätt att låta snabba CPU:er undvika blockerande operationer som väntar på ett långsamt nätverk.

- a. I stället för att skicka bytes använder NIO channels. Hur fungerar detta? (2p)
- b. Vad gör man i stället för att blockera på read i en while loop? (2p)

4. **Sockets:** Socket-API:n innehåller en rad funktioner. Anta att det rör sig om en TCP socket. Beskriv följande funktioners uppgift. Samt koppla dem till TCP protokollet.(2p)

- a. bind()
- b. accept()

4. **OSI:** är en lageruppdelad nätverksmodell där varje lager har sina specifika uppgifter. Vissa protokoll och teknologier mappar väldigt bra mot just det lager de ligger i med avseende på deras ansvar. De nedan uppräknade är sådana. Vilka är dessas absolut viktigaste uppgifter?(2p)

- a. IPv6
- b. Ethernet

5. **Trådar:**

- a. Beskriv när synchronisering behövs.(2p)
- a. Förklara begreppet deadlock, när det uppstår.(1p)
- b. Beskriv hur man undviker deadlock.(1p)
- b. Beskriv utförligt tre olika sätt att ”få” ett resultat från en tråd.(3p)

6. **Serialisering:**

- a. Vad innebär Serialisering?(1p)
- b. Hur skiljer sig Serialisering åt mellan JAVA och C++?(2p)

7. **Applikationsprotokoll:** Ett väderlekssystem bestående av sensorer som levererar data till en väderleksserver. Klienter skall kunna: Registrera sig hos servern. Få en lista med serverns sensorer. Begära väderleksdata, från önskade sensorer, av servern. Avregistrera sig. Till detta behövs ett protokoll. I framtiden kan nya versioner av protokollet komma att skapas och servern ska då kunna hantera alla versioner. Protokollet skall vara språkneutralt.

- a. Ett antal meddelanden måste definieras i protokollet. Meddelandena skall vara i binärt format (ej klartext). Exempel på ett meddelande är *Unregister*. Detta skickas av klient för att avregistrera sig hos servern. Ge exempel på fyra meddelanden, utöver *Unregister*, samt vad dessa gör.(2p)
- b. Tre fält måste vara gemensamma för alla meddelanden. Vad skall dessa fält innehålla?(3p)
- c. Beskriv så att en, för dig okänd, programmerare kan implementera *Unregister*.(4p)
- d. Val av transportprotokoll är fritt men valet påverkar delar av designen för protokollet. Välj ett av transportprotokollen, motivera valet och beskriv hur det påverkar designen av protokollet.(1p)