Tentamen
Nätverksprogrammering
Del 1

2018–03–14, 8.00–13.00

Tillåtna hjälpmedel för denna del av tentamen: *inga*. Kurslitteratur och andra hjälpmedel för del 2 av tentamen skall förvaras på golvet bredvid bordet eller vid salens vägg.

Denna tentamen i kursen Nätverksprogrammering består av två delar – en del som innehåller frågor av teoretisk/principiell/utredande karaktär och en del som innehåller praktiska programmeringsuppgifter. Detta är del 1. När du löst uppgifterna i denna del av tentamen lämnar du in din lösning i det vita tentamensomslaget varvid du erhåller del 2 av tentamen tillsammans med ett nytt, färgat, tentamensomslag som skall användas vid inlämning av din lösning på del 2 av tentamen.

För godkänt betyg på tentamen krävs sammanlagt minst 20 poäng på tentamen, varav minst 8 poäng på vardera deltentamen. För högre betyg krävs mer, så gör så många uppgifter du kan.

---

1. Under kursen har vi använt en nätverksmodell som består av fyra lager: applikationslagret, transportlagret, internetlagret och fysiska lagret. För applikationslagret och transportlagret: beskriv kortfattat vad lagret ansvarar för och ge ett exempel på ett protokoll i det lagret.

   (2p)

2. Vad är kapplöpning i samband med datorprogram? När uppstår det? Hur undviker man kapplöpning i ett javaprogram?

   (3p)

3. Ett REST-api är ett vanligt sätt att göra data tillgängligt på internet, t.ex. när en webb-applikation läser/skriver data från/till en server.

   a) Ett RESTfull api bygger på ett annat protokoll, vilket?

   (1p)

   b) Hur anger man vilken operation, t.ex. läsa eller skriva, som ett REST-anrop ska utföra.

   (1p)

   c) Hur vet man om en REST-operation lyckades eller misslyckades.

   (1p)

   d) Hur identifieras olika resurser, t.ex. vilket objekt man vill läsa/skriva, vid REST-anrop?

   (1p)

4. I kursen förekommer begreppet *multicast*.

   a) Beskriv kortfattat vad det är och hur det fungerar.

   (2p)

   b) Vid multicast förekommer begreppet *Time To Live* (TTL). Vad är detta och vad är motivet till att man infört det?

   (1p)

---

5. Vilka av följande sex påståenden är korrekta? Rätt svar get *+0,5p*, felaktigt svar *-1p*. Lämnas inget svar ger det *0p*. Hela uppgiften kan inte ge en negativt poäng.

   a) Strängen `"http://cs.lth.se/edaf65/några exempel.pdf"` är korrekt formaterad enligt RFC2396, d.v.s den kan användas som värde till `new URL(String url)`.

   b) Alla uri:er är även url:er.

   c) Om ett program läster från en TCP-socket på en port kan den samtidigt skicka UDP-packet från samma port.

   d) En XML-fil kan innehålla binär data, d.v.s. base64-encoding behövs inte.

   e) Alla XML-filer refererar till en DTD.

   f) PHP kan användas av en webb-applikation för att exekvera kod i webbläsaren.

(3p)

6. När man använder strömmar i form av en TCP-uppkoppling är det ofta fördelaktigt med att använda en *buffrad* ström.

   a) Nämn en fördel med buffring av strömmar.

(1p)

   b) Buffring kan också medföra problem om man inte ser upp. Beskriv ett scenario där buffring kan medföra att kommunikationen mellan två datorer via TCP inte fungerar. Hur löser man problemet i Java?

(2p)

7. Ida Idler har skrivit ett multitrådat javaprogram i vilket hon vill kunna tillfälligt pausa exekveringen av en tung bakgrundstråd. Detta vill hon göra eftersom tråden annars konsumerar allt för mycket CPU-tid. Hon har därför skrivit nedanstående klass för kontrollera bakgrundstråden.

```
public class PauseControl {
    private boolean paused = false;

    public void setPause(boolean state) {
        paused = state;
    }

    public void pause() {
        while(paused);
    }
}
```

Klassen fungerar så att bakgrundtråden regelbundet anropar metoden `pause()`. Eftersom attributet `paused` normalt är `false` kommer inget att utföras i anropet och bakgrundstråden fortsätter exekvera. Om någon annan tråd däremot ändrar värdet på attributet, genom att anropa metoden `setPause()`, kommer bakgrundstråden att stanna i while-satsen tills attributets värde återställs.

   a) Idas lösning fungerar dock inte särskilt bra. Vad har hon gjort för misstag?

(1p)

   b) Använd dina kunskaper om trådsynkronisering i Java för att skriva om klassen `PauseControl` ovan så att den fungerar som det var tänkt.

(1p)

Slut på del 1 – lämna in och hämta ut del 2!