

# Prov i moment C om datorintroduktion och Unix

Version 3

31 augusti 2023

## 1 Inledning

Detta prov är praktiskt och genomförs i grupper om två studenter, eller enskilt, och inför en labbhandledare på en någorlunda avskild plats i datorsalarna. Provet består av de 7 uppgifterna i avsnitt 5 nedan och ska lösas i LTU:s Unix-miljö (Ubuntu).

## 2 Förberedelse innan provet

Innan labbhandledare kontaktas och provet tas ska gruppen<sup>1</sup> göra följande.

1. Hitta en avskild<sup>2</sup> plats längst bak i salen med gott om plats för tre personer och logga där in i LTU:s Unix-miljö, antingen med en av salens datorer eller en dator någon i gruppen äger (om skärmen är tillräckligt stor).
2. Öppna två (2) jämnstora terminalfönster som kör *Ubuntu*<sup>3</sup>. I det ena fönstret ska uppgifterna lösas och i det andra ska detaljer ur en fil finnas synliga under provet (se provuppgift 2 nedan).
  - Viktigt: För att det ska vara lätt för såväl studenter som labbhandledare att läsa text på skärmen ska höjden på stora bokstäver som visas i fönstren vara minst 5 mm.

---

<sup>1</sup>Instruktionen är skriven som att ett par tar provet och tar man provet enskilt så får man förstås göra allt själv.

<sup>2</sup>Ingen ska kunna tjuvtitta, och helst inte heller kunna tjuvlyssna, under provet.

<sup>3</sup>Med egen dator kan man här behöva logga in två gånger för att få två fönster.

3. Den aktuella arbetskatalogen ska i båda fönster vara hemmakatalogen, den plats i filträdet man automatiskt hamnar på då man loggar in
4. Det är inte tillåtet med några hjälpmedel under provet.
5. När det gäller högre utbildning så innebär arbete i grupp inte bara att gemensamt lösa en uppgift. Det innebär också att hjälpa varandra ordentligt förstå alla delar av lösningen så var och en är väl insatt i allt.  
Det är inför detta prov tillåtet – och definitivt rekommenderat – att gemensamt gå igenom alla uppgifterna flera gånger innan det är dags att ta provet.
6. I provuppgift 5 ska ett av programmen `xclock` (med flaggan `-update 1`), `xcalc` eller `xeyes` startas. Se i förväg till att du kan starta, stoppa och återstarta dessa tre i fönstren. (Detta kräver tekniskt sett att `s k X11-forwarding` fungerar.)

### 3 Under provet

Under provet gäller följande.

- Under provet ska studenterna växelvis gå igenom och göra provuppgifterna 1-7 nedan i ordning. Medan de gör detta ska de själva berätta om vad de gör och varför.  
OBS! Uppgifterna beror delvis på varandra, dvs en senare uppgift kan bero på nåt som gjorts i en tidigare.
- Båda studenterna ska visa labbhandledaren att de kan lösa provuppgifterna med hjälp av Unix-kommandon och under tiden även besvara frågor från labbhandledaren om vad/hur de gör som troliggör att de vet vad de gör och varför.
- Alla uppmaningar i uppgifter, som t ex att skriva ut något, ska utföras av Unix-systemet som resultat av ett eller flera Unix-kommandon.

## 4 Resultat

För godkänt ska alla provuppgifter lösas och alla frågor besvaras nöjaktigt. Det ska vara klart för labbhandledaren att båda i gruppen förstår varje steg i allt som görs.

Om gruppen inte klarar att lösa uppgifterna och/eller inte kan berätta/förklara väl så får gruppen retur och måste ta om provet senare. Om gruppen inte blir godkänd efter tre (3) returer så blir gruppen underkänd och får istället återkomma nästa gång kursen ges för att då ta provet igen.

Hela gruppen, de båda studenterna, blir godkända, får retur eller blir underkända.

## 5 Provuppgifter

Vissa detaljer, som behövs för att lösa uppgifterna, saknas i denna text. Detaljerna fås av labbhandledaren som del av uppgift 2.

### Uppgift 1

- 1.1 Skapa en katalog med namnet hage i hemkatalogen, gå ner i den och skriv ut absolutadressen till den aktuella arbetskatalogen.

### Uppgift 2

- 2.1 Kopiera filen `/nfs/courses/SRT/d0015e/zip/X.zip` med bibehållet namn till arbetskatalogen, där X är en siffra som fås av labbhandledaren.

- 2.2 Skriv kommandot `unzip X.zip` och ange det lösenord som labbhandledaren ger och visa att arbetskatalogen, utöver X.zip, nu även innehåller filerna `dikt.txt` och `detaljer.txt`.

I `detaljer.txt` finns ett antal detaljer som ska användas när resten av provuppgifterna görs. Detaljerna känns igen på att de är ord skrivna med

`STORA BOKSTÄVER INOM RAM` i denna text.

- 2.3 Skriv ut `detaljer.txt` i det ena av terminalfönstren och ha innehållet synligt medan ni löser uppgifter i det andra fönstret under resten av provet.

- 2.4 Skapa en katalog med namnet `KATALOGNAMN` taget ur `detaljer.txt` i arbetskatalogen. Flytta sen, med ett enda kommando, de två filerna `X.zip` och `dikt.txt` till `KATALOGNAMN`.
- 2.5 Gå ner i `KATALOGNAMN`, byt namn på `dikt.txt` till `FILNAMN.txt` (där `FILNAMN` tas ur `detaljer.txt`) och lista alla filer i arbetskatalogen så diverse information visas inklusive senaste tidpunkt filerna ändrats.

### Uppgift 3

Innehållet i `FILNAMN.txt` saknar rubrik och namn på diktaren.

- 3.1 Lägg till rubrik och namn på diktaren, dvs visa hur ni med valfria kommandon, men inte någon editor, ser till att det sen i katalogen `KATALOGNAMN` finns en (möjligen ny) fil med namnet `FILNAMN.txt` som har textraderna:
- (a) `RUBRIK` (ur `detaljer.txt`) inom citationstecken (rad 1 alltså),
  - (b) tomrad,
  - (c) ordet "av" följt av `NAMN` (ur `detaljer.txt`),
  - (d) tomrad och därefter (fr o m rad 5)
  - (e) alla textrader ur själva dikten.
- 3.2 Skriv, när ni är klara, ut (precis) de 20 första raderna ur `FILNAMN.txt`, så labbhandledaren tydligt ser att ni klarat av uppgiften.
- 3.3 Radera eventuella tillfälliga filer ni skapat när ni gjort denna uppgift. (Ta dock inte bort någon av `X.zip`, `detaljer.txt` eller `FILNAMN.txt`.)

### Uppgift 4

I dikter kallas styckena "strofer".

- 4.1 Använd en pipe (ett "rör") och koppla ihop två kommandon till en kommandosekvens som skriver ut (precis) de rader ur `FILNAMN.txt` som bildar diktens första strof. Granska `FILNAMN.txt` och den första strofen som finns i den filen för att veta vilka rader som ska skrivas ut.

- 4.2 Använd en pipe (ett "rör") och koppla ihop två kommandon till en kommandosekvens som skriver ut antal rader i `FILNAMN.txt` som innehåller sekvensen med bokstäver i `SÖKORD` (ur `detaljer.txt`).
- 4.3 Vilken veckodag var den 1 januari år `ÅR` (ur `detaljer.txt`), dvs det år dikten gavs ut?

## Uppgift 5

- 5.1 Skriv ut vilken dator du är inloggad på. Logga sen in på en av
- `stuterm1.sts.ltu.se`,
  - `stuterm2.sts.ltu.se` och
  - `stuterm3.sts.ltu.se`
- som ni just nu inte är inloggade på och skriv åter ut vilken dator ni är inloggade på.
- 5.2 Skriv ut ert användarnamn (dvs den användare ni loggat in med), lista alla som är inloggade och logga sen ut.
- 5.3 Starta `PROGRAM` (ur `detaljer.txt`) så det kör i bakgrunden och Unix-prompten omedelbart skrivs ut igen.
- 5.4 Lista därefter alla egna processer och peka särskilt ut processen som kör `PROGRAM` samt dess PID.
- 5.5 Stanna och återstarta processen som kör `PROGRAM`, och visa att programmet inte reagerar/fungerar medan processen är stoppad, men fortsätter fungera så fort processen åter startats.
- 5.6 Ge ett kommando som avslutar processen som kör `PROGRAM`.

## Uppgift 6

- 6.1 Kopiera `/nfs/courses/SRT/d0015e/kod/spel.c` till den katalog hage ni skapade i uppgift 1. Gå till katalogen `'hage` och ge ett kommando som visar vilken timme det är.

- 6.2 Använd en editor (valfri men t ex `nano` som i förberedelserna) och ändra den övre intervallgränsen i `spel.c` till entalssiffran i födelseåret för en av er i gruppen adderat med vilken timme det är. Exempel: Om klockan är 16:35 så skulle någon född 2001 ändra till  $16 + 1 = 17$ .
- 6.3 Visa sen att ni kan "kompilera" och spela spelet.

## Uppgift 7

- 7.1 Flytta alla filer från katalogen `KATALOGNAMN` till närmast överliggande katalog (den som innehåller `KATALOGNAMN`).
- 7.2 Gå till katalogen `hage` som skapades i uppgift 1, radera katalogen `KATALOGNAMN`, och radera alla filer.
- 7.3 Gå till hemkatalogen, radera katalogen `hage` och visa avslutningsvis att hemkatalogen inte längre innehåller någon katalog `hage`.

Slut på provuppgifterna.