

Grundläggande Operativsystem 1DV415 - Seminarie 1

Marcus Wilhelmsson
marcus.wilhelmsson@lnu.se

5 november 2013

Introduktion

Detta seminarie behandlar följande:

1. Introduktion till hårdvara, mjukvara och operativsystem
2. Processer och trådar

Alla svar måste motiveras.

Läshänvisningar:

- Föreläsning 1 och 2
- Operating Systems - Internals and design principles
 - Kap. 1–4
- Operativsystem - Teori och praktiskt handhavande
 - Kap 2, 5–8
- Internet

Frågor

1. Ett operativsystem administrerar och tar bara hand om hårdvara. Sant eller falskt?
2. Vilka är operativsystems främsta uppgifter?
3. Vad kunde utgöra gränsen för vilken storlek och vilka funktioner ett program på t.ex. 1950-talet kunde utföra?
4. Operativsystemen TSS, Multics och CP/CMS hade alla en speciell funktion inbyggd i operativsystemet som kunde ses som en mindre revolution. Vad innebar denna funktion?
5. Under 1970-talet började datorer i allt större utsträckning kommunicera med varandra över TCP/IP-baserade nätverk. Nämn ett problem som detta kan medföra samt hur man kan motverka det.

6. Vilken aspekt hos moderna operativsystem och PC-datorer, som blev populär med hjälp av Apple Macintosh, gjorde dem lättare att lära sig och förstå?
7. En server kan inte vara en klient. Sant eller falskt?
8. Nämn två fördelar med mjukvara som är öppen källkod.
9. Vilken typ av system skulle du välja för att övervaka temperaturen i ett kärnkraftverk? Varför?
10. Vilken operativsystemskomponent utför var och en av följande uppgifter?
 - (a) Skriva till hårddisken.
 - (b) Bestämna vilken process som ska köras närmast.
 - (c) Bestämna var i minnet en ny process ska placeras.
 - (d) Organisera filerna på hårddisken.
 - (e) Göra så att processer kan skicka data till varandra.
11. Vilka är de speciella egenskaperna hos ett monolitiskt operativsystem?
12. Vad är skillnaden mellan lagerbaserade operativsystem och sådana med mikrokärna?
13. Termerna "process" och "program" är synonyma. Sant eller falskt?
14. En process kan hamna i "blockerat" läge när den väntar på att något ska ske. Ge exempel på två saker som en process kan vänta på i detta läge.
15. Vad gör operativsystemet för att se till att en process inte kan ha monopol på en processor och aldrig släppa den till andra processer?
16. Vad är skillnaden mellan en process som är vaken (awake) och en som sover (asleep)?
17. När kan det vara bättre att sätta en process i *suspend* istället för att avbryta den?
18. Varför bör ett operativsystem minimera tiden som krävs för att utföra en context switch?
19. Förklara vad ett avbrott (interrupt) är.
20. Vad finns det för alternativ till avbrott? Varför används detta så sällan?
21. Nämn en nackdel med signaler jämfört med IPC (InterProcess Communication).
22. Vad är fördelen med att köra en multitrådad applikation på ett multiprocessorsystem jämfört med att köra den på ett system med bara en CPU?
23. Hur är följande tillstånd lika/olika?
 - (a) Väntande (waiting)
 - (b) Blockerad (blocked)
 - (c) Sovande (sleeping)
24. Om vi använder oss av många-till-en-trådmappning (även kallat user level-trådar) blockeras hela processen av operativsystemet även om det egentligen bara är en tråd som behöver blockeras. Hur kommer detta sig?
25. När är det mer effektivt att använda sig av kernel level-trådning jämfört med user level-trådning?