



### Försättsblad Prov Original

Kurskod	DT011G	Provkod	T106	Tentamensdatum	2018 - 08 - 28
Kursnamn	Datateknik GR (A), Operativsystem introduktionskurs				
Provnamn	Tentamen				
Ort	Sundsvall				
Termin	H18				
Ämne	Datateknik				

# Tentamen

Operativsystem introduktionskurs GR (A), 7,5 hp, DT011G

Operativsystemsteori GR (A), 7,5 hp, DT141G

Jimmy Åhlander

2018-08-28

Tid	5 timmar
Hjälpmedel	Inga
Maxpoäng	50
Krav för godkänt	För godkänt betyg på tentamen fordras betyg E, eller högre.

Preliminära betygsgränser	A	B	C	D	E
	90 %	80 %	70 %	60 %	50 %


**Instruktioner** Skriv **tydligt**. Oläsliga svar erhåller inga poäng även om svaret är korrekt.

Besvara uppgifterna på separata svarpapper. Skriv endast på framsidan av svarpappret. Påbörja varje uppgift på ett nytt blad. Svaret på en uppgift kan alltså sträcka sig över flera blad, men maximalt en uppgift, med underuppgifter, får besvaras på ett givet blad.

Använd inte penna med röd skrift.

Markera sidnummer och din kod längst upp till höger på *varje blad*, se nedanstående figur. Numrera bladen *i efterhand* så att uppgifterna kommer i en logisk ordningsföljd.

*Skriv aldrig ut ditt namn eller personnummer.*



Anonymitetskod: 000-00	Sidnr: 1
1. a)	

**Instruktioner till tentamensvakt** Försättsbladet och studentens svar skannas automatiskt och får *inte* häftas ihop.

Studenten får efter avlagd tentamen medta dessa instruktioner med tillhörande uppgifter. Väljer de att inte göra detta ska dessa instruktioner med tillhörande uppgifter kasseras. Endast de ohäftade svarpappren och försättsbladet skickas tillbaka.

### Uppgift 1 (5 x 1p)

Definiera och förklara följande grundläggande begrepp inom ramen för operativsystem.

- Sekventiell åtkomst.
- Avbrott (*interrupt*).
- Skal.
- Filsystem.
- Tråd.

### Uppgift 2 (3 x 2p)

Sidersättningsalgoritmer tillämpas i operativsystem som nyttjar virtuellt minne och demand paging. Logiska sidor läses in till fysiska ramar och kastas bort utefter processernas behov. Givet **tre fysiska ramar** och referenssträngen **R2 R1 W4 W0 W1 R0 R3 R4 W1**, där R representerar en läsoperation och W representerar en skrivoperation.

- Hur många sidfel uppstår om sidersättningsalgoritmen Second Chance (SC) tillämpas? *Visa hur du kommit fram till ditt svar.*
- Hur många sidfel uppstår om sidersättningsalgoritmen First-In First-Out (FIFO) tillämpas? *Visa hur du kommit fram till ditt svar.*
- Ger algoritmerna samma resultat? Varför? Kommer det alltid att vara sant för alla referenssträngar? *Motivera.*

### Uppgift 3 (1 + 3 + 2p)

Synkroniserad exekvering av processer är en förutsättning för att de ska kunna samarbeta med varandra. Begrepp relaterade till synkronisering är den kritiska sektionen och det kritiska sektionsproblemet.

- Vad är den kritiska sektionen?
- Vad är det kritiska sektionsproblemet? Vilka krav ställs på eventuella lösningar?
- Vad innebär racetillstånd (*race condition*)?

### Uppgift 4 (1 + 2 + 2p)

När paging introduceras så ändras den effektiva åtkomsttiden till primärminnet. Anta att du har ett 64-bit operativsystem där åtkomsttiden utan paging är 200 ns. Systemet har ingen *translation lookaside buffer* (TLB).

- Vad blir den effektiva åtkomsttiden när paging introducerats?
- Varför ändras den effektiva åtkomsttiden till primärminnet när paging används?

För att förbättra åtkomsttiden introducerar vi en TLB till systemet. Efter att ha kört ett antal tester kan vi visa att bufferten har en träffsäkerhet på 80 % och en åtkomsttid på 10 ns. Med en TLB får vi en effektiv åtkomsttid  $t$  motsvarande följande formel:

$$t = p \times (t_t + t_m) + (1 - p) \times (t_t + t_m + t_m)$$

där  $p$  motsvarar sannolikheten att hitta något i bufferten,  $t_t$  tiden för en buffertåtkomst, och  $t_m$  tiden för en minnesåtkomst.

- Vad blir den effektiva åtkomsttiden med en TLB i detta fall?

### Uppgift 5 (1 + 2,5 + 1,5 + 1p)

Processer är centrala för ett operativsystems funktion. Operativsystem lagrar metadata för processer i strukturer kallade processkontrollblock (PCB). Besvara följande frågor relaterade till processer.

- Vad är skillnaderna mellan ett program och en process?
- Rita ett tillstånds- och övergångsdiagram (*state-transition diagram*) för de olika lägena en process generellt kan befinna sig i. Förklara även kortfattat de olika lägena.
- I UNIX-lika operativsystem, vad innebär zombieläget för processer? Vad är den teoretiska risken med för många zombies i operativsystemet?
- Oavsett operativsystem så arbetar processer vanligen i en *I/O-burst cycle*. Vad innebär det?

### Uppgift 6 (2 + 4 + 1p)

Deadlocks kan uppstå när vissa givna kriterier är uppfyllda. I ett visst datorsystem råder ett specifikt scenario, se Tabell 1. Systemet har ingen preemption eller deadlockförhindrande algoritm. Allokering av resurserna kan antas ske slumpartat. Samtliga resursinstanser är odelbara.

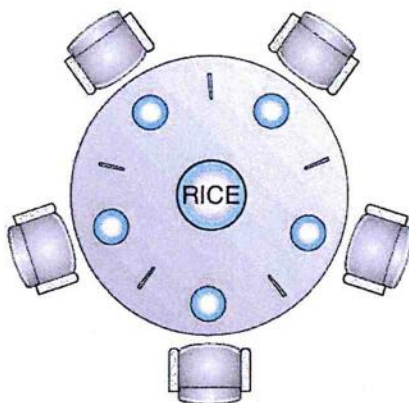
Tabell 1. Det maximala antalet instanser en process kan efterfråga per resurs.  
Maximalt tillgängligt antal resursinstanser inom parentes.

Process	R <sub>0</sub> (1)	R <sub>1</sub> (?)	R <sub>2</sub> (2)	R <sub>3</sub> (2)
P <sub>0</sub>	1	2	0	1
P <sub>1</sub>	1	1	0	1
P <sub>2</sub>	0	0	1	0
P <sub>3</sub>	0	2	2	0

- Hur många instanser av resursen R<sub>3</sub>, om någon alls, krävs minimalt för att förhindra uppkomsten av en deadlock? *Motivera ditt svar.*
- Ange och beskriv kraven för att en deadlock ska kunna uppstå.
- Inom kontexten för deadlocks omnämns ibland *strutsalgoritmen*. Vad innebär det?

### Uppgift 7 (4p)

Förklara de ätande filosofernas problem (*eng. dining philosophers problem*). Se även Figur 1. Ge även minst ett förslag på hur problemet kan lösas som uppfyller alla kraven för *the critical section problem*.



Figur 1. The dining philosophers problem.

### Uppgift 8 ( 1 + 1 + 1 + 2 + 2p )

CPU-schemaläggning används för att avgöra vilken process som ska få exekvera närmast på en processor. I följande scenario väntar ett antal processer i redo-kön för att få exekvera. Processerna kan antas ha ankommit i kön vid samma tid.

Process	Uppskattad burst-tid (ms)	Ankomst i kön (#)	Prioritet
P2	5	1	7
P3	10	2	8
P0	6	3	8
P1	10	4	8

Processerna kan exempelvis schemaläggas med följande tre algoritmer: *First-Come, First-Served (FCFS)*, *Shortest-Job-First (SJF)*, och en specialiserad prioritetsschemaläggning.

Den specialiserade prioritetsschemaläggningen utgår i första hand ifrån prioriteten där ett större värde representerar en högre prioritet, i andra hand efter uppskattad burst-tid där ett lägre värde har en högre prioritet, och i tredje hand efter köplats där ett lägre värde har en högre prioritet.

- Vad blir medelväntetiden om processerna schemaläggs med FCFS? *Visa dina beräkningar.*
- Vad blir medelväntetiden om processerna schemaläggs med SJF? *Visa dina beräkningar.*
- Vad blir medelväntetiden om processerna schemaläggs med den beskrivna prioritetsschemaläggningen? *Visa dina beräkningar.*
- Förklara hur schemalägningsalgoritmen *Round Robin (RR)* fungerar.
- Vad bör beaktas vid valet av tidskvantumets storlek i RR?

### Uppgift 9 ( 1p )

Vad innebär Direct Memory Access (DMA) och vad används det för?

### Uppgift 10 ( 3 x 1p )

Tre olika typer av fragmentering kan uppstå i sekundärminnet när blockallokering används. Förklara hur de olika typerna uppstår och hur det leder till problem.

### Checklista

- ✓ Kryssa i alla påbörjade uppgifter i försättsbladet.
- ✓ Numrera bladen så att uppgifterna kommer i numerisk och logisk ordningsföljd.

## Lexikon

EN	SV
Access.....	Åtkomst
Atomic.....	Atomär eller atomisk
Burst.....	Sprint
Context switch .....	Kontextbyte
Contiguous .....	Sammanhängande
CPU-scheduling .....	Processorschemaläggning
Critical section.....	Kritisk sektion
Cycle.....	Cykel
Displacement, offset .....	Förskjutning
Deadlock.....	Dödläge eller baklås
Demand.....	Krav, efterfrågan, behov
Enhanced.....	Förbättrad
File permissions.....	Åtkomstkontroll
Fragmentation.....	Fragmentering
Frame .....	Ram
Heavy-weight process .....	Tungviktarprocess ( <i>ej vedertagen översättning</i> )
Indexed allocation.....	Indexerad allokering
Interrupt.....	Avbrott
Kernel mode.....	Kärnläge
Linked allocation .....	Länkad allokering
Memory-management techniques .....	Minneshanteringstekniker/-strategier ( <i>ej vedertagen översättning</i> )
Memory-management unit.....	Minneshanteringsenhet
Mode.....	Läge
Multilevel feedback queue.....	Flernivååterkopplingskö
Page.....	Sida
Page fault .....	Sidfel
Page replacement algorithms .....	Sidersättningsalgoritmer
Page table.....	Sidtabell
Paging.....	Paging ( <i>rimlig översättning saknas, se Page</i> )
Pointer.....	Pekare
Preemption .....	Möjlighet till avbrott eller expropriation
Preemptive.....	Förebyggande eller spärrande
Pure.....	Ren eller äkta
Race condition.....	Tävlingstillstånd
Segmentation .....	Segmentering
Semaphore .....	Semafor
Sequential .....	Sekventiell
State .....	Läge eller tillstånd
System call .....	Systemanrop
Thrashing.....	Thrashing ( <i>rimlig översättning saknas</i> )
Transition .....	Övergång
User mode.....	Användarläge
Quantum (time), time slice .....	Tidskvantum