



TENTAMEN **TDDD82** Säkra mobila System (Systemprogramvara)

Datum: **2015-06-11**

Tid: 14 - 18

Lokal: TER2

Jourhavande-lärare:

- Mikael Asplund (telefonnr: **0700 895 827**) – OBS! endast tillgänglig via telefon pga fotskada

Hjälpmedel: Behövs inte.

Poängantal: 34p

Betyg: 3, 4 and 5:

U:	- 16 p
3:	17 - 22 p
4:	23 - 28 p
5:	29 - 34 p

INSTRUCTIONS:

Write your anonymous ID number on each sheet of paper that you hand in. Further, pages should only contain **answer to one question per page** (answers to sub-questions can be on the same page). You are asked to only answer on the front page of the paper. **Sort all the sheets** that you hand in, ordered in question number.

Your answers can be presented in Swedish or English.

Make sure your answers include motivations and are presented precisely. A correct answer **without any motivation will not be given any credits**. Incorrect answers embedded in a partially correct one reduce the points given for that answer. Points will not be given to answers that cannot be read due to bad handwriting.

Hints: Try to dispose of your time on each question in proportion of the assignment points. In those cases where you are in doubt about the question, write down your interpretation and your assumptions, and answer the question based on the interpretation. Figures can be of help when describing but should be accompanied by a text description.

Results are reported no later than 1st July, 2015.

Good luck!

Simin Nadjm-Tehrani
Examiner TDDD82, Systems Software

Q1:

- a) Semafor fungerar för att åstadkomma ömsesidig uteslutning endast om operationerna **wait** och **signal** implementeras som *atomiska operationer*. Beskriv två olika metoder för att få dessa operationer att bli atomiska, och för varje metod ange en nackdel.

(4 poäng)

- b) Betrakta följande pseudokod för programmet som består av processerna P1 och P2:

```
Process P1 {
  X = 1;
  Y = X-2;
  If Y > 1 then
    X = X - 1
  else
    X = X + 2
}

Process P2 {
  X = 3;
  Y = X-1;
  If Y < 1 then
    X = X + 1
  else
    X = X - 2
}
```

Ange 2 spår som hör till programmet och ett spår som *inte* hör till programmet.

(3 poäng)

- c) Ett system som ska visa positionen av varje lagmedlem i en räddningstjänstgrupp på en karta i varje medlems smarttelefon har implementerats med hjälp av en central server. Positionen för varje medlem uppdateras hos en databas som underhålls på serversidan. Den här frågan gäller tre processer på klientsidan. Två processer *Update* och *Transmit* delar den gemensamma resursen sändbuffer enligt producer-consumer mönstret och använder en semafor för att implementera ömsesidig uteslutning. När en medlem har rört sig tillräckligt långt att positionen anses behöva uppdatering (avstånd från tidigare lagrad punkt är större än en tröskel) så finns det anledning att placera ett meddelande med ny position i sändbuffern. Den tredje processen *Refresh* avgör helt enkelt om en uppdatering behövs genom att hämta nuvarande position från GPS på mobila enheten och avgöra när villkoret är uppfyllt (enheten har rört sig tillräckligt mycket). Presentera pseudokoden för de tre processerna *Refresh*, *Update*, och *Transmit* med tillhörande semafor operationer och korrekt initialisering av semaforer (både för ömsesidig uteslutning och villkorssynkronisering).

(6 poäng)

- d) Förklara vad menas med "race condition" i parallell processers sammanhang.

(2 poäng)

- e) Tillståndet i ett system där flera processer delar flera resurser kan beskrivas mha ett resursallokeringsdiagram. Betrakta följande beskrivning av ett systems tillstånd där det finns tre parallella processer (P_A , P_B och P_C) samt tre delade resurser (R_1 , R_2 , R_3). Sammanlagt finns det 8 instanser av R_1 och R_3 var, samt 6 instanser av R_2 .

Resurs(↓) Process (→)	P _A	P _B	P _C
R ₁	2	1	0
R ₂	1	0	1
R ₃	0	2	2

- I. Beskriv systemets nuvarande tillstånd med hjälp av en resursallokeringsdiagram. (1 poäng)
- II. Anta att process B kommer med en begäran av (1, 1, 0) resurser i nuvarande tillstånd. Anta vidare att processerna har initialt angett sina max behov enligt följande matris:

Resurs(↓) Process (→)	P _A	P _B	P _C
R ₁	4	2	1
R ₂	1	1	2
R ₃	0	2	2

Med hjälp av Bankirsalgoritmen avgör om systemet ska acceptera att allokeras resurserna enligt P_B's begäran.

(2 poäng)

Q2:

- a) Ta ställning till följande utsagor (sant eller falsk), och motivera ditt svar med hänvisning till definitioner av termer, eller med hjälp av förklarande exempel.
- 1) All applications that do not tolerate QoS variations are non-elastic.
 - 2) Diffserv is preferred over Intserv since it has a lower overhead for route set up.

(4

poäng)

- b) Betrakta en mobil applikation som ska serva tågresenärerna med realtidsinformation om ankomsttider, förseningar, ersättningar av tågresor med bussresor osv. För varje krav som anges nedan avgör om det är ett funktionellt eller icke funktionellt (också kallad extra-funktionellt) krav. Motivera ditt svar!

1. Programmets serversida skall ta emot varje tågs nuvarande position var tionde minut.
2. Servern ska kunna hantera minst 100 tågs uppdateringar, och minst 10000 resenärers uppkoppling inom X resp. Y sekunder.

3. Om en resenär har betalt för en första klass biljett och servern är överbelastad får resenären servas före andra med resor som köpta inom andra klassen.

(3 poäng)

Q3:

- a) Presentera två olika felscenarier inom det *distribuerade* systemet som beskrivs under Q2 b) genom att använda kategorierna "crash, omission, timing, Byzantine, ..." och för varje felscenario presentera en metod som kan användas för att hantera felet enligt IFIP Working Group 10.4 terminologi.

(4 poäng)

- b) Beskriv en arkitektur inom området distribuerade system som *inte* baseras på klient-server modellen.

(1 poäng)

Q4:

Använd begreppen från IFIP Working Group 10.4 för att analysera felkälla-felyttring-haveri i detta sammanhang, och klassificera felkällan som "permanent/transient/intermittent".

"In May 2015, a gossip blog based in the Dominican Republic called Remolacha published a disturbing video of what it said was a self-parking car accident. A group of people stand in a garage watching and filming a grey Volvo XC60 that backs up, stops, and then accelerates toward the group. It smashes into two people, and causes the person filming the video with his phone to drop it and run.

The main issue, said [Volvo spokesperson Johan] Larsson, is that it appears that the people who bought this Volvo did not pay for the Pedestrian detection functionality, which is a feature that costs more money.

The Volvo XC60 comes with City Safety as a standard feature, however this does not include the Pedestrian detection functionality, said Larsson. The City Safety system kicks in when someone is in stop-and-go traffic, helping the driver avoid rear ending another car while driving slowly, or under 30 mph."

(4 poäng)

Glossary: English to Swedish

atomic – atomisk (oavbrytbar)	request – begäran
availability – tillgänglighet	response time – responstid
avoid – undvika	safety – säkerhet
bandwidth – bandbredd	scheduler – schemaläggare
broker – medlare	security – säkerhet
clock drift – klockdrift	serialisable - serialiserbar
concurrency – samtidighet	shaping - formning
consistency – konsistens	shared memory – gemensamt minne
deadlock – låsning (baklås)	starvation – svält
delay – fördröjning	sufficient condition- tillräckligt villkor
deliberate – avsiktlig	synchronisation – synkronisering
dependability – pålitlighet	system call – systemanrop
diversity – mångfald	thread - tråd
error – felyttring	throughput – genomströmning
event – händelse	trace – spår
failure – haveri	transparency – transparens
fault – felkälla	trust – tillit
fault tolerance – feltolerans	validation – validering
forecast – förutse	verification - verifiering
inheritance – arv	vulnerability - sårbarhet
integrity – dataintegritet	
interleavings – sammanflätningar	
interoperability – interoperabilitet	
intrusion – intrång	
latency - latens	
maintainability – reparerbarhet	
malicious - illvillig	
middleware - mellanvara	
mutual exclusion – ömsesidig uteslutning	
necessary condition – nödvändigt villkor	
non-functional property – ickefunktionell egenskap	
omission – utelämnande	
performance – prestanda	
preemptible – avbrytbar	
prevent – förebygga	
quality of service – tjänstekvalitet	
race condition – kapplöpningstillstånd	
release - släppning	
reliability – tillförlitlighet	
redundancy – redundans	
replica – kopia	
replication – replikering	

Swedish – English

arv - inheritance	samtidighet - concurrency
atomisk (oavbrytbar) – atomic	sårbarhet - vulnerability
avbrytbar - preemptible	schemaläggare - scheduler
avsiktlig - deliberate	serialiserbar - serialisable
bandbredd - bandwidth	släppning – release
begäran - request	spår - trace
dataintegritet - integrity	svält - starvation
felkälla - fault	synkronisering - synchronisation
feltolerans - fault tolerance	systemanrop - system call
felyttring - error	tillförlitlighet - reliability
fördröjning - delay	tillgänglighet - availability
förebygga - prevent	tillit - trust
formning - shaping	tillräckligt villkor - sufficient condition
förutse - forecast	tjänstekvalitet - quality of service
gemensamt minne - shared memory	tråd - thread
genomströmning - throughput	transparens - transparency
händelse - event	undvika - avoid
haveri - failure	utelämnande - omission
ickefunktionell egenskap - non-functional property	validering - validation
illvillig - malicious	verifiering - verification
interoperabilitet - interoperability	
intrång – intrusion	
kapplöpningstillstånd – race condition	
klockdrift - clock drift	
konsistens - consistency	
kopia - replica	
låsning (baklås) - deadlock	
latens - latency	
mångfald - diversity	
medlare - broker	
mellanvara - middleware	
nödvändigt villkor - necessary condition	
ömsesidig uteslutning - mutual exclusion	
pålitlighet - dependability	
prestanda - performance	
redundans - redundancy	
reparerbarhet - maintainability	
replikering - replication	
responstid - response time	
säkerhet - safety	
säkerhet - security	
sammanflätningar - interleavings	

