Tentamen

Operativsystem, 5p
25 mars 1998

- Skrivtid: 9 – 15
- Hjälpmedel: Miniräknare
- Maxpoäng: 40 (Gräns för 3: **20** p (50%), 4: **26** p (65%), 5: **32** p (80%))
- Börja varje uppgift på *nytt* blad och fyll i alla uppgifter på *varje* blad.
- Resultat kommer upp på anslagstavlan så snart jag rättat klart.
  (Kommer även att finnas tillgängligt på WWW.)
- Ni kan hämta tentorna på mitt rum så snart resultet kommit upp.
- Fråga om någonting är oklart!

  Lycka Till!

Uppgift 1 (4 + 4 p)

Om att man har följande minnesaccesser (refererade adresser) som skulle kunna komma från ett litet program:

`10 108 672 72 356 436 78 112 82 396 408 618 360 412`

Antag att programmet exekverar i ett system med en sidstorlek på 100 bytes och med 400 bytes tilldelat primärminne och att alla frames är tomma från början.

(a) Hur många page faults ger följande algoritmer?

- LRU
- FIFO
- Optimal algoritm (OPT)
- Second Chance (även kallad klockalgoritm)

Visa också (för varje algoritm) vilka sidor som finns i minnet vid tidpunkten för minnesreferensutdragets slut.

(b) Nämn en fördel och en nackdel för varje algoritm i (a).

Uppgift 2 (5 p)

För att kunna implementera ett operativsystem som inte påverkas av felaktiga program krävs visst hårdvarustöd. Nämn 3 sådana hårdvarumekanismer och förklara varför de behövs och hur man kan använda dem.

Uppgift 3 (2 + 2 p)

(a) Vad är *garbage collection*? När och varför används det?

(b) Beskriv en metod för *garbage collection*.

Uppgift 4 (2 + 2 + 2 p)

(a) Vilka villkor måste gälla för att ett system ska befinna sig i dödläge?

(b) Beskriv en metod för att undvika dödläge.

(c) Om vi har ett system med fyra likadana resurser som delas av tre processer som var och en som mest behöver två resurser. Kan systemet hamna i dödläge? Motivera ditt svar!

Uppgift 5 (4 p)

När uppträder *page faults*? Beskriv vad operativsystemet gör när ett *page fault* inträffar.

Uppgift 6 (3 p)

I samband med skyddsfrågor (*protection*) så pratar man om *access lists* och *capability lists*, beskriv med några rader vad dessa är och eventuella skillnader och likheter.

Uppgift 7 (3 p)

Vilka är skillnaderna mellan threads och traditionella processer i Unix-miljö.

Uppgift 8 (2 p)

Förklara skillnaden mellan mjuka och hårda länkar i traditionella Unix-system genom att beskriva hur de realiseras i filsystemet.

Uppgift 9 (4 + 1 p)

The Cigarette-Smokers Problem. Consider a system with three *smoker* processes and one *agent* process. Each smoker continuously rolls a cigarette and then smokes it. But to roll and smoke a cigarette, the smoker needs three ingredients: tobacco, paper, and matches. One of the smoker processes has paper, another has tobacco, and the third has matches. The agent has an infinite supply of all three materials. The agent places two of the ingredients on the table. The smoker who has the remaining ingredient then makes and smokes a cigarette, signaling the agent on completion. The agent then puts out another two of the three ingredients, and the cycle repeats.

(a) Using semaphores, write algorithms to synchronize the agent and the smokers.

(b) Ge en kortfattad beskrivning vad som eventuellt skulle behöva ändras i din lösning om fler rökare, agenter eller ingredienser tillåts.