CHALMERS
EXAMINATION / TENTAMEN

Course code/kurskod	Course name/kurskod			
EDA093	Operating Systems			
Anonymous code Anonym kod		Examination date Tentamensdatum	Number of pages Antal blad	Grade Betyg
EDA093-0015-GST		15-08-2023	7	3

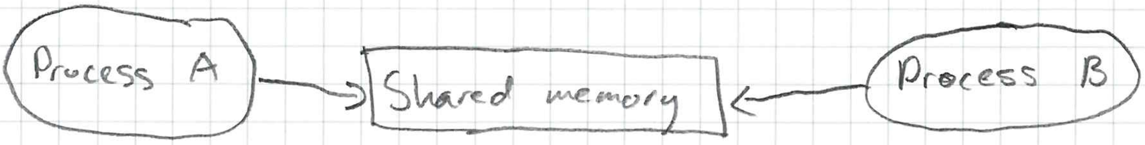
* I confirm that I've no mobile or other similar electronic equipment available during the examination.
Jag intygar att jag inte har mobiltelefon eller annan liknande elektronisk utrustning tillgänglig under eximinationen.

Solved task Behandlade uppgifter No/nr		Points per task Poäng på uppgiften	Observe: Areas with bold contour are to completed by the teacher. Anmärkning: Rutor inom bred kontur ifylles av lärare.
1	X	7	
2	X	6	
3	X	8	
4	X	10	
5	X	4	
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
Bonus poäng			
Total examination points Summa poäng på tentamen		35	

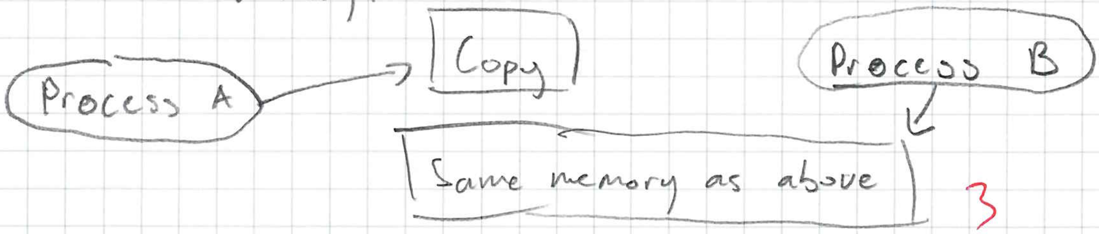
5

CHALMERS	Anonymous code Anonym kod EDA093-0015-GST	Points for question (to be filled in by teacher) Poäng på uppgiften (ifylles av lärare)	Consecutive page no. Löpande sid nr 1 Question no. Uppgift nr 1.

a) Copy-on-write (COW) enables different processes to share memory. Only when the shared memory will be modified will the memory be copied.



A modifies the shared memory, OS copies the memory.



3

Benefits of COW are that memory will only be copied when neccessary, meaning less computation which will be needed by the OS and more free memory.

b)

Logical

	0
A	1
C	2
	3
	4
E	5
	6
D	7 ??
	8
B	9

Page table

0	
1	5
2	3
3	
4	
5	8
6	
7	9
8	
9	0

bits?

Frame

B	0
	1
	2
C	3
	4
A	5
	6
	7
E	8
D	9

1

c) 1. Handle page fault interrupt
   a) Illegal memory access → Crash
   b) Page not in frame → Continue with step 2.

2. Remove unwanted page (if there is any) to disk. Update the page table.
3. Find and put wanted page into the frame

4. Update page table.

3

5. Resume the process which caused the page fault.

Answer only one question on this page. Do not write on the back of this paper
Behandla endast en uppgift på detta blad. Skriv ej på baksidan

5

CHALMERS	Anonymous code Anonym kod EDA093-0015-GST	Points for question (to be filled in by teacher) Poäng på uppgiften (ifylles av lärare)	Consecutive page no. Löpande sid nr 2 Question no. Uppgift nr 1.

d) Yes, because then pages are only loaded into memory if and when they are needed.

and that decreases PF?

0

Answer only one question on this page. Do not write on the back of this paper
Behandla endast en uppgift på detta blad. Skriv ej på baksidan

5

CHALMERS	Anonymous code Anonym kod EDA093-0015-GST	Points for question (to be filled in by teacher) Poäng på uppgiften (ifylles av lärare)	Consecutive page no. Löpande sid nr 3 Question no. Uppgift nr 2.

a) Amdahl's law describes the speedup of a program of which $S \cdot 100\%$ runs sequentially and N cores / threads are used:

$$\text{speedup} \leq \frac{1}{S + \left(\frac{1-S}{N}\right)}$$

3

b) 1. Scheduler interrupt: process A will be paused

2. Store A's register values, PC, page table etc into memory.

3. Retrieve B's register values, PC, page table etc from memory and load it into its corresponding unit.

4. Resume process B.

The second context switch is the same as above, with the small change of A and B. This time B will be paused, the same operations done as before, only this time first on B, then A.

3

c) User-level threads Kernel-level threads

+ Fast context switches

Ø

Answer only one question on this page. Do not write on the back of this paper
Behandla endast en uppgift på detta blad. Skriv ej på baksidan

CHALMERS

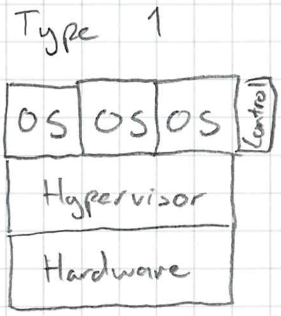
Anonymous code / Anonym kod: EDA093-0015-GST

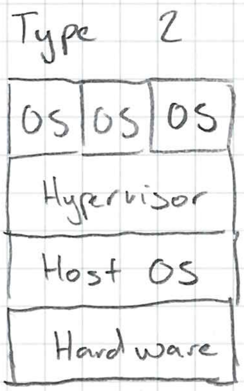
Points for question (to be filled in by teacher) / Poäng på uppgiften (ifylles av lärare)

Consecutive page no. / Löpande sid nr: 4

Question no. / Uppgift nr: 3.

a)





In type 1 the hypervisor runs directly on HW. Type 2 runs on a host OS.

3

NO

c) A hypervisor has a shadow page table for each VM. Each VM has a page table, and the shadow page table bridges the gap between a VM's page table and the actual frames in physical memory.

The two methods are guest-induced page faults and hypervisor-induced page faults.

2

d) Incremental snapshots are those were only the modifications since the last complete snapshot are stored. This reduces the memory footprint significantly.

3

Answer only one question on this page. Do not write on the back of this paper
Behandla endast en uppgift på detta blad. Skriv ej på baksidan

CHALMERS	Anonymous code Anonym kod EDA093-0015-GST	Points for question (to be filled in by teacher) Poäng på uppgiften (ifylles av lärare)	Consecutive page no. Löpande sid nr 5 Question no. Uppgift nr 4.

```
int   thread_sem;

DataQueue   queue;
main () {
     queue  = data_queue_create();
     thread_sem=create_semaphore (2);

     pthreads  A, B, C;

     A = p_threads_create (*a_method);

     B = p_threads_create(* b_method);

     C = p_threads_create(* c_method);

     shed_yield();

}

a_method () {
     while ( true  {
     thread_sem. wait ();

     files  = check_files ();

     if  (files  ==  null)  {
         thread_sem. signal(); shed_yield();
     }
      else  {

          for  (file : files) {
              queue .add (file);
          }
          thread_sem. signal();
          shed_yield();

     }
    }
}

b_method() {
    while (true) {
        thread_ sem. wait ();
        element  =  queue.pop();
        if (element != null) {
              do_computation (element);
        }
        thread_sem. signal ();
        shed_yield();
    }

}
```

waiting for them?

16

Answer only one question on this page. Do not write on the back of this paper
Behandla endast en uppgift på detta blad. Skriv ej på baksidan

CHALMERS	Anonymous code	Points for question (to be filled in by teacher)	Consecutive page no. Löpande sid nr 6
	Anonym kod EDA093-0015-GST	Poäng på uppgiften (ifylles av lärare)	Question no. Uppgift nr 4~~8~~

```
c_method() {
    thread_sem.wait();
    int max_size = get_size_from_user();
    thread_sem.signal(); shed_yield();
    while(true) {
        thread_sem.wait();
        if (queue.size  > max_size) {
            queue.remove(queue.size - max_size);
        }
        thread_sem.signal();
        shed_yield();

    }
}
```

Assumption: Since DataQueue is thread-safe, it is assumed that no races are possible in regards of the queue.

Answer only one question on this page. Do not write on the back of this paper
Behandla endast en uppgift på detta blad. Skriv ej på baksidan

5

CHALMERS	Anonymous code Anonym kod EDA093-0015-GST	Points for question (to be filled in by teacher) Poäng på uppgiften (ifylles av lärare)	Consecutive page no. Löpande sid nr 7 Question no. Uppgift nr 5.

Peterson's algorithm: (Same for thread B but vice versa)

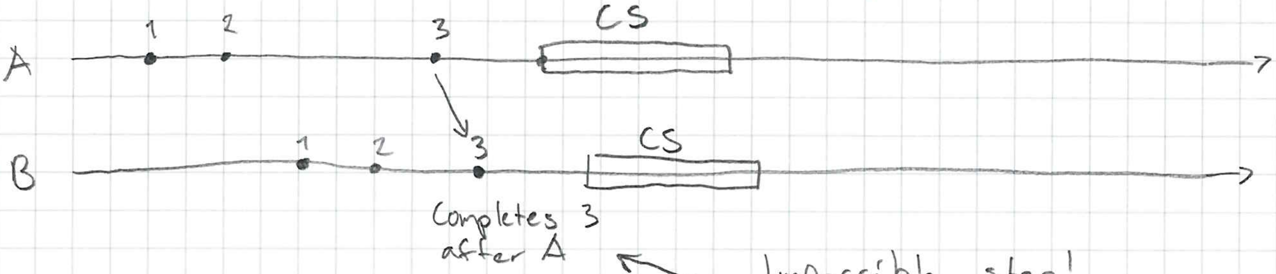
```
boolean  flagA, flagB   // init false
int      turn, idA, idB

repeat {   // Thread A
1.      flagA ← true
2.      turn ← idA   ← B
3.      while (flagA and turn == idA) do nothing
            critical section
4.          flagB ← false
            remainder section
}
```

which case is this flagA false or turn = idB?

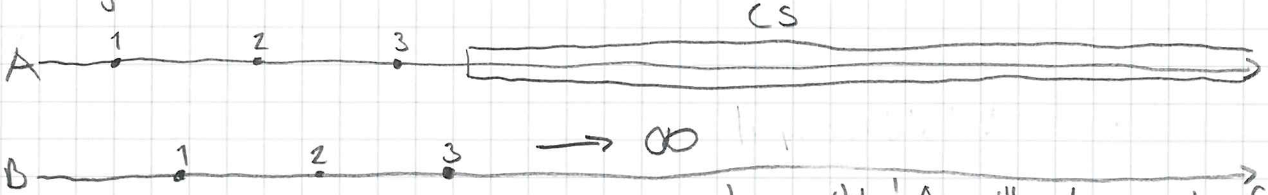
4

Mutual exclusion:



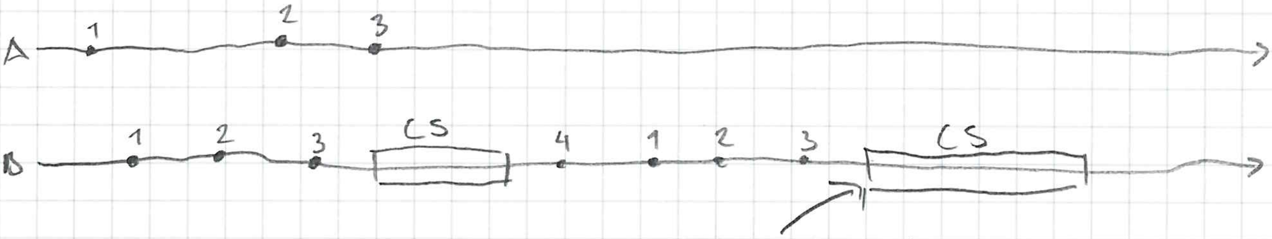
Completes 3 after A ← Impossible step!

Progress:



Impossible! A will step out of CS, and then B can continue.

Fairness:



Impossible! B cannot starve A.