

Godkendelsesopgave 5 i “Styresystemer og multiprogrammering”

Generelt

Denne ugeopgave skal afleveres senest torsdag den 7. marts 2008 klokken 6:00. Den kan løses af grupper på op til 3 personer. Besvarelsen af opgaven vil resultere i enten 0, $\frac{1}{2}$ eller 1 point. Pointene uddeles efter følgende retningslinjer:

- 0 point: besvarelsen har flere store mangler.
- $\frac{1}{2}$ point: besvarelsen opfylder i store træk kravene men har flere mindre mangler.
- 1 point: en god besvarelse der kun har få eller ingen mangler.

Det er en betingelse for at gå til eksamen på kurset at man har opnået mindst 4 point i alt, og at mindst fem ugeopgaver har fået mindst $\frac{1}{2}$ point.

Besvarelsen skal indleveres elektronisk via kursushjemmesiden på ISIS. I skal bruge arbejdsgruppe funktionaliteten i ISIS når I afleverer for en gruppe. I skal oprette en arbejdsgruppe og tilføje medlemmerne til gruppen. Herefter kan I aflevere som en samlet gruppe under 'Aflevering af opgaver'. Systemet sørger ikke for et unikt gruppenavn, så brug følgende skabelon til at navngive Jeres gruppe:

`efternavn1-efternavn2-efternavn3`

Skulle dette ikke være nok til at sikre at jeres gruppenavn er unikt, kan I anvende fornavne, fødselsdage, eller tilfældige tal til at sikre unikhed.

Besvarelsen skal ske ved aflevering af en enkelt fil. Brug 'zip' eller 'tar.gz' til at samle flere filer. Filnavnet skal have følgende format:

`efternavn1-efternavn2-efternavn3-Hold<h>-G<n>.<endelse>`

hvor <n> er opgavenummeret og <h> er holdnummeret for den instruktør som sidst rettede Jeres opgave.

Jeres rapport skal være i PDF eller ASCII format, for at lette instruktørernes rettearbejde. Opgavenummer og navne på gruppemedlemmer skal fremgå tydeligt af første side i rapporten.

Afleveringen skal indeholde én rapport på 1-3 sider der dokumenterer hver delopgave. Kravene til dokumentation er specificeret i hver opgave. I skal også huske at kommentere Jeres kildetekst så den let at forstå.

Tjekliste for aflevering:

- **Opgavenummer og navne på gruppemedlemmer skal fremgå tydeligt af første side i rapporten.**
- Brug arbejdsgruppe funktionaliteten i ISIS.
- Følg navngivningsskabelonen for arbejdsgrupper og afleveringsfil (se ovenfor).

- Rapport i PDF eller ASCII format.
- Brug 'zip' eller 'tar.gz' til at samle alle filer i én fil.
- Kontroller at I har opfyldt kravene til opgaveaflevering for hvert delspørgsmål.

Introduktion

I denne opgave skal I dels arbejde praktisk med sidetabeller (*eng.: page tables*), dels skal I løse en teoretisk opgave om sideerstatning (*eng.: page replacement*).

G5.1: Sidetabel

I denne opgave skal I implementere en datastruktur for en to-niveau sidetabel som beskrevet i lærebogens afsnit 8.5 (figur 8.14). Det logiske adresseformat antages at være som i lærebogen side 298. Hvis I arbejder på en 64-bit maskine kan I ignorere de mest betydende bits, samt udelade adresser større end $2^{32} - 1$.

Datastrukturen for sidetabellen skal kunne tilgås via to funktioner som I skal implementere:

```
/*
Sætter 'page' ind i sidetabellen 'pt'. Den logiske adresse
afgør hvor i sidetabellen 'page' skal placeres. Hvis
'page==NULL' slettes siden fra sidetabellen.
*/
void ptable_set(ptabel_t *pt, void *laddr, void *page);

/*
Returnerer pegeren ('page') som er blevet indsat med 'ptable_set'
for den pågældende logiske adresse 'laddr'. Hvis den logiske
adresse ikke eksisterer, returneres 'NULL'.
*/
void *ptable_get(ptabel_t *pt, void *laddr);
```

De pegere som indsættes og udtages af sidetabellen, vil normalt indeholde information om siden – f.eks. fysisk adresse, beskyttelse osv. Dette skal I ikke bruge i denne opgave og I kan derfor ignorere det. I skal altså kun lave en datastruktur som oversætter en logisk adresse til en adresse (peger), og som bruger strukturen for to-niveau sidetabeller.

Den vedlagte kildetekst indeholder en skabelon for Jeres implementation ('ptabel.c'). I denne finder I også et eksempel på hvordan man laver bitoperationer på pegere i C. Faktisk kan man ikke lave bitoperationer på pegere uden konvertering.

I skal selv designe en test af Jeres implementation. Opgavebesvarelsen for dette delspørgsmål skal indeholde:

- Kildeteksten i filen 'ptabel.c'.
- Rapport: Beskrivelse af hvordan I har implementeret sidetabellen samt testen. Testresultater!

G5.2: Sideerstatning

I denne opgave betragtes en computer der anvender virtuelt lager (*eng.: virtual memory*) med sidedeling og *demand paging*. Der er 4 sidepladser (*eng.: page frames*) nummereret fra 0 til 3. Systemet anvender *second chance* sideudskiftningsalgoritmen. På tidspunktet T har sidetabellen følgende indhold:

Sideplads	0	1	2	3
Side	5	8	16	4
Reference bit	1	0	1	1

og det næste offer er sideplads 2 (altså undersøges sideplads 2 som den første næste gang der skal udskiftes en side).

Der genereres nu referencer til følgende sider (startende med tidspunktet T+1 og sluttende med tidspunktet T+12):

Tid T+	1	2	3	4	5	6	7	8	9	10	11	12
Side	5	6	16	8	5	9	10	6	7	8	3	7

Angiv på skematisk og overskuelig form for hver reference om der sker sidefejl (*eng.: page fault*), hvilken sideplads der udskiftes og hvilke ændringer der sker i sidetabellen. Angiv hvor mange sidefejl der sker i alt.

Besvarelsen skal indgå i Jeres rapport.