

Oppgavesettet består av 3 sider

## WESTERDALS OSLO ACT

### INNLEVERING #2

### PG3300 Programvarearkitektur

Leveranseinfo:

- Skal gjøres i grupper på 2-3 personer ( gjerne samme som for innlevering #1).
  - Leveres senest søndag 16. november 2014, klokka 23:55.
  - Leveres via it's learning.
  - Kode skal skrives i C#, og kompilere og kjøre i Visual Studio Ultimate 2013.
  - UML diagrammer kan være Visual Studio eller bildeformat (bildeformat inkluderer screenshot).
  - Tekstdokumentasjon skal være i doc, docx eller pdf format.
- 

## Oppgave 1: Forklaring av Multithreading

Forklar følgende begreper i forbindelse med multithreading: (med utgangspunkt i C#, om det håndteres forskjellig for forskjellige språk)

### Oppg. 1 a) Race conditions

Hva er race conditions?

Hvordan kan disse oppstå?

### Oppg. 1 b) Locks

Hva er locks?

Hvordan benytter vi locks i C#?

### Oppg. 1 c) Deadlocks

Hva er deadlocks?

Hvordan kan disse oppstå?

---

## Oppgave 2: The Cookie Bakery

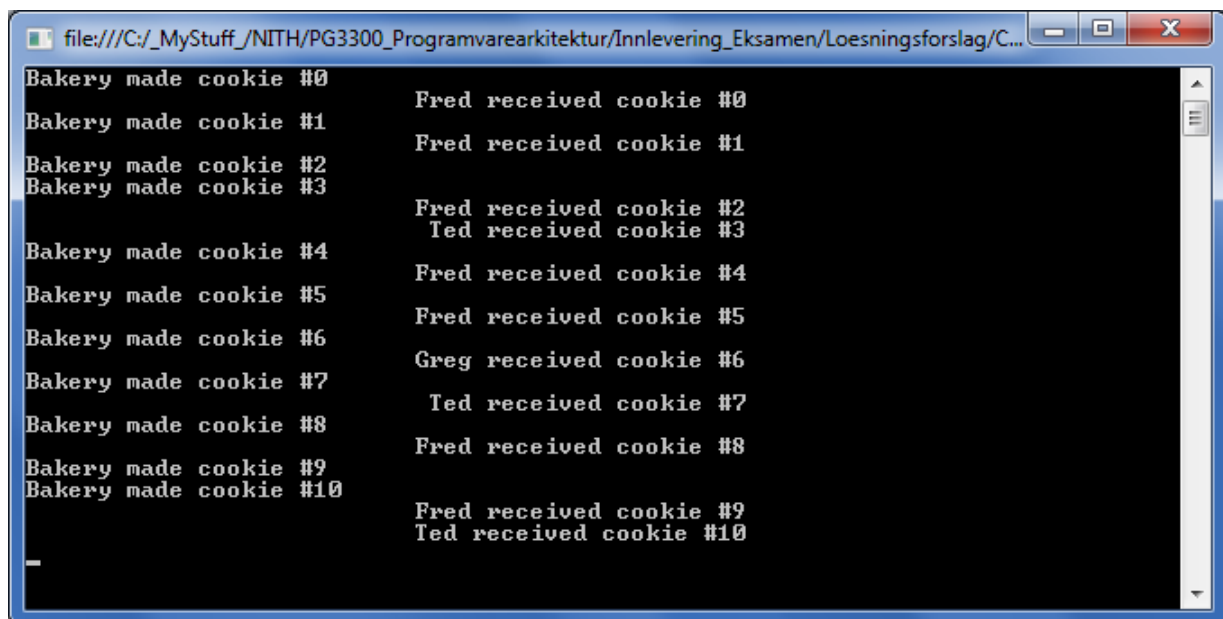
"The Cookie Bakery" er et lite bakeri med tre faste stamkunder: De glupske herrene Fred, Ted og Greg.

The Cookie Bakery lager kun cookies (of course), og selv om produksjonen kan være nokså lav (minimum et dusin cookies om dagen, utover det velger dere selv) hindrer ikke det Fred, Ted og Greg fra å storme mot disken jævnt og trutt for å se om det ligger nye cookies der! (Rekkefølgen ferdige kaker legges ut for salg i trenger ikke være lik rekkefølgen de stekes i.) Men uansett hvor samtidig herrene kaster seg mot disken, er det jo bare én som kan få tak i hver cookie, ikke sant?

Bakeriet putter sine cookies i en kurv på disken etterhvert som de er ferdig bakt (for eksempel hvert 667. millisekund). De holder åpent til de har bakt en dagsvorte og denne er solgt unna.

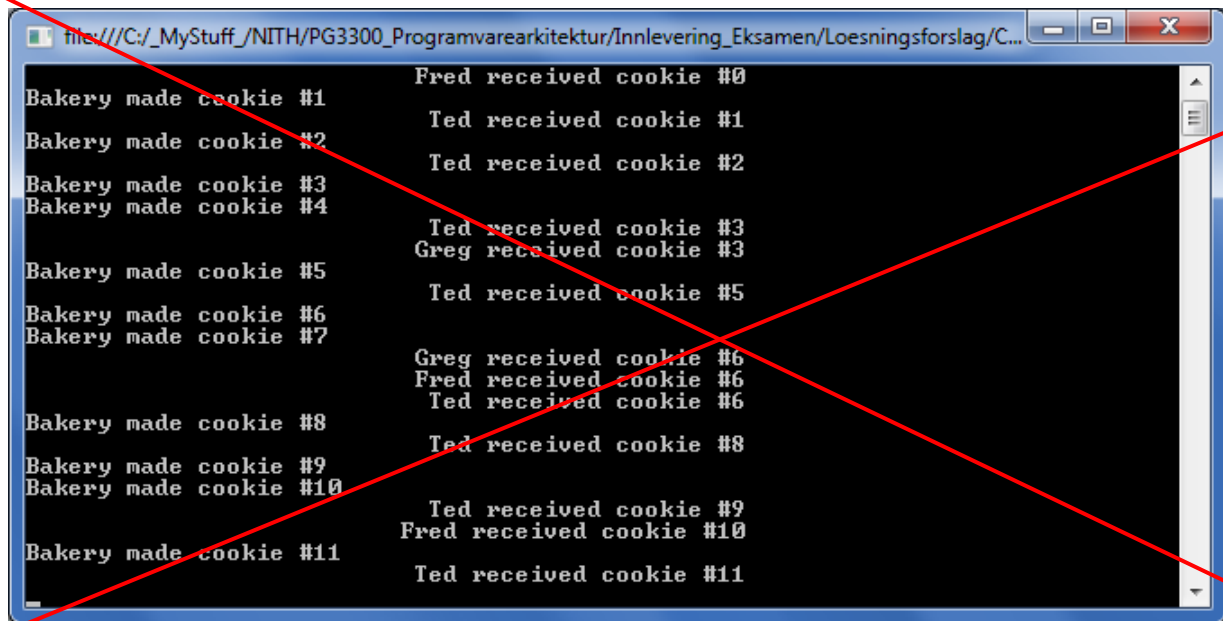
Fred, Ted og Greg prøver alle å få tak i cookies ved jævne mellomrom! Implementer funksjonalitet som gjør de i stand til å prøve å plukke opp kaker samtidig (multithreading), men sørg også for at koden er trådsikker slik at bare én kan få fingrene i hver kake som kommer. (Teknisk: Bakeriet selger kakene gjennom en egen metode – `"SellCookieTo(Customer customer)"` eller tilsvarende. Fred, Ted og Greg kaller denne metoden jevnlig gjennom hver sin thread, for eksempel en gang per 1000 millisekunder - velg et tall som er litt høyere enn for bakingen.)

Eksempel på kjøring – i dette tilfellet er Fred helt klart den beste cookie knabberen:



```
file:///C:/_MyStuff/_NITH/PG3300_Programvarearkitektur/Innlevering_Eksamen/Loesningsforslag/C...
Bakery made cookie #0      Fred received cookie #0
Bakery made cookie #1      Fred received cookie #1
Bakery made cookie #2      Fred received cookie #2
Bakery made cookie #3      Ted received cookie #3
Bakery made cookie #4      Fred received cookie #4
Bakery made cookie #5      Fred received cookie #5
Bakery made cookie #6      Greg received cookie #6
Bakery made cookie #7      Ted received cookie #7
Bakery made cookie #8      Fred received cookie #8
Bakery made cookie #9      Fred received cookie #9
Bakery made cookie #10     Ted received cookie #10
_
```

Som i skjermbildet under skal det **ikke** se ut! Flere herrer kan nemlig ikke spise den samme kaka! (Ref. cookie #3 og cookie #6.)



```
file:///C:/_MyStuff/_NITH/PG3300_Programvarearkitektur/Innlevering_Eksamen/Loesningsforslag/C...
Bakery made cookie #1      Fred received cookie #0
Bakery made cookie #2      Ted received cookie #1
Bakery made cookie #3      Ted received cookie #2
Bakery made cookie #4      Ted received cookie #3
Bakery made cookie #5      Greg received cookie #3
Bakery made cookie #6      Ted received cookie #5
Bakery made cookie #7      Greg received cookie #6
Bakery made cookie #8      Fred received cookie #6
Bakery made cookie #9      Ted received cookie #6
Bakery made cookie #10     Ted received cookie #8
Bakery made cookie #11     Ted received cookie #9
                          Fred received cookie #10
                          Ted received cookie #11
```

#### Oppg. 2 a) Dokumentering

Diskuter muligheter innad i gruppen, og begrunn valgene deres (bl.a. hvorfor dere velger variant A fremfor B, der hvor dere kommer opp med flere muligheter). Dette skal danne grunnlaget for et dokument (doc, docx eller pdf) der dere gjør rede for valgene og prosessen deres. Husk at dette er en del av sensuren: Det er viktig at dere forklarer løsningen deres for sensor!

Med andre ord: I dette tekstvedlegget, sørg for å få fram hvorfor akkurat deres løsning er så bra, dvs. vis (med ord og evt. figurer) at dere har gjort gode valg for løsningen deres! (Som for innlevering #1, men antakeligvis er det færre valg, og derfor mindre å skrive denne gangen.)

Om dere har benyttet UML, legg ved diagrammene (og eventuell beskrivelse) her.

Sørg også for å begrunne eventuelle design patterns dere benytter.

#### Oppg. 2 b) Implementering

Implementer "The Cookie Bakery" som en multithreaded applikasjon der Fred, Ted og Greg kniver om å få tak i cookies!

Merk – som for innlevering #1: Tenk god struktur, ikke nødvendigvis hva som hadde vært mest praktisk for et så lite prosjekt i arbeidslivet.

- Slutt på oppgaven -