|  |
| --- |
| Kungliga Tekniska Högskolan |
| Utkast experimentplanering |
|  |
|  |
| **Sadok Habibi Dalin, 19940519-6974. Alexander Barosen, 19940205-5710** |
| **2015-12-08**  **Sadokd@kth.se, Abarosen@kth.se**  **Ingenjörskunskap och ingenjörsrollen ICT II1304** |

Innehållsförteckning

[1. Resultat och effektmål 3](#_Toc436684190)

[1.1 Resultatmål 3](#_Toc436684191)

[1.2 Effektmål 3](#_Toc436684192)

[2. Förstudie 3](#_Toc436684193)

[2.1 Uppgiften och systemet som skall studeras 3](#_Toc436684194)

[2.2 Vilka parametrar beror systemet av? 3](#_Toc436684195)

[2.3 Vilka parametrar kan varieras? 3](#_Toc436684196)

[2.4 Hur beror parametrarna av varandra? 3](#_Toc436684197)

[2.5 Studiens karaktär 3](#_Toc436684198)

[3. Experimentplan 3](#_Toc436684199)

[3.1 Vad som ska mätas och hur? 3](#_Toc436684200)

[3.2 Insamling och analys av data 4](#_Toc436684201)

[3.3 Beskrivning av de enskilda experimenten 4](#_Toc436684202)

[4. Experimentuppställningen 4](#_Toc436684203)

[4.1 Resurskrav 4](#_Toc436684204)

[4.2 Vilka osäkerhets/felkällor finns? Hur stor inverkan kan de ha på resultaten och hur hanterar du det? 5](#_Toc436684205)

[4.3 Validering av experimentuppställningen 5](#_Toc436684206)

[4.4 Villkorlig förändring av experimentplanen 5](#_Toc436684207)

[4.5 Hur induktiva respektive deduktiva metoder kan användas 5](#_Toc436684208)

[5. När är målen uppfyllda? När är experimenten färdiga? 5](#_Toc436684209)

[5.1 När är de identifierade målen uppfyllda 5](#_Toc436684210)

[5.2 När är experimenten färdiga 5](#_Toc436684211)

[Appendix A: Kravanalys 6](#_Toc436684212)

# 1. Resultat och effektmål

## 1.1 Resultatmål

Det finns tre huvudsakliga resultatmål med uppgiften:

* Att skriva en experimentplanering.
* Att skriva en teknisk rapport som innehåller en experimentell utvärdering av fyra algoritmer för användarstyrd minneshantering.

## 1.2 Effektmål

Det finns två huvudsakliga effektmål med uppgiften:

* Träna på grunder i kvantitativ metodik.
* Lära sig genomföra experiment samt analysera och presentera resultat från experiment.
* Skapa sig en överblick över vilka algoritmer som passar i olika situationer.

# 2. Förstudie

## 2.1 Uppgiften och systemet som skall studeras

Vi har visat att vi har förstått uppgiften och systemet vi ska studera genom att utförligt analysera och dokumentera de givna kraven på systemet och utvärderingen.

## 2.2 Systemets parametrar

Om ett system är synonymt med implementationen av var och en av algoritmerna så beror varje system av den stokastiska variabeln X, antalet generade nya händelser, som är likformigt fördelat på intervallet [0, N] och den stokastiska tiden increment(). Increment() genererar ett decimaltal likformigt fördelat i intervallet [0, Y]. Varje system beror också av köns initiala tillstånd d.v.s. hur många händelser som kön initialt innehåller.

## 2.3 Parametrarnas variation

X kan sättas till ett konstant värde. Alternativt kan N sättas till ett valfritt värde. Increment() kan också sättas till ett konstant värde eller generera ett tal i intervallet [0, Y] där . I kön kan ett valfritt antal händelser fler än 0 placeras initialt.

## 2.4 Parametrarnas beroende

Parametrarna beror inte direkt av varandra. X och increment() kan dock generera tal i liknande mönster om båda använder samma slumptalsgenerator.

## 2.5 Studiens karaktär

Studien är en kvantitativ studie d.v.s. utvärderingen bör utmynna i ett resultat som enkelt kan ges ett numeriskt värde.

# 3. Experimentplan

## 3.1 Vad som ska mätas och hur?

Det som ska mätas är tiden det tar att göra en insättning i de olika köerna. Detta ska göras med hjälp utav tidsstämplar. Innan ett element placeras i kön tar man en tidsstämpel x. När elementet är placerat i kön tas en tidsstämpel y och dokumenteras.

## 3.2 Insamling och analys av data

Det som ska samlas in är tiden det tar att göra en insättning i de olika köerna. Tiderna ska vara i decimalform och ska anges i ms. En mängd med tider kommer att samlas in för varje algoritm i bästa-, värsta- och normalfall.

För varje mängd kommer den kortaste och den längsta tiden tas ut. Dessa tider är bästafalls- respektive värstafallstiden. Normalfallstiden tas fram genom att beräkna medelvärdet av alla tider i varje mängd. Bästa-, värsta- och normalfallstiderna för alla algoritmer kommer sedan att jämföras för att utvärdera prestandan.

## 3.3 Beskrivning av de enskilda experimenten

Varje experiment utförs fem gånger för fem olika slumptalsfrön för att få statistiskt valid mätdata. Varje experiment utförs flera gånger för var och en av algoritmerna.

Experiment ”Shemaläggningskö”:

* Köns initiala tillstånd: [0, 0, 0, 0, 0, 1, 1, 1, 1, 1… 40, 40, 40, 40, 40].
* Antalet genererade nya processer per uttagning: 1.
* De genererade processerna tilldelas en likformigt fördelad prioritet i intervallet [0, 40].
* Inga processer läggs tillbaka efter att de har tagits ut.

Körning med konstant mängd processer/trådar. Detta görs genom att låta processerna som tas ut ur kön, endast och alltid genererar en ny process/tråd för insättning. Totala antalet processer blir oförändrat. Eftersom totala mängden element i kön är konstant kan alla mättider enklare jämföras med de teoretiska uppskattningarna.

Experiment ”Händelselista”:

* Köns initiala tillstånd: [0.1, 0.2, 0.3…, 10].
* Antalet genererade nya händelser per uttagning: 1.
* Increment() genererar decimaltal likformigt i intervallet [0, 3.0].

Körning med en lista som har en konstant storlek. Genom att låta alla händelser alltid och endast generera en ny händelse för varje uttagning för blir totala mängden element i listan oförändrad.

Eftersom det är konstant mängd händelser i kön så kan man enklare jämföra mätvärden med de förväntade teoretiska värdena.

Denna kombination av experiment utförs för att undersöka prestandan hos de olika prioritetsköerna för olika belastning.

# 4. Experimentuppställningen

## 4.1 Resurskrav

För att utföra experimenten behövs flera resurser:

* Ett system att köra på. Alla experiment kommer utföras på en dator med följande specifikationer:
  + CPU: Intel Xeon X3470 2.93 Ghz
  + Minne: 15 GB RAM.
  + OS: Linux 3.13.0-68-generic.
  + Kompilator: GCC 4.8.4,
* En C-kompilator såsom GCC eller MinGW.
* Implementationen av de fyra algoritmerna.
* Ett sätt att mäta tid.
* Ett sätt att mäta minneskomplexitet.

## 4.2 Vilka osäkerhets/felkällor finns? Hur stor inverkan kan de ha på resultaten och hur hanterar du det?

I vår studie finns både interna och externa felkällor:

* Slumptalsvarians. Eftersom både increment() och X genereras från slumptalsgeneratorer så kan dåligt valda slumptalsfrön och få körningar leda till förutsägbara mönster i resultaten. Således kan dålig hantering av slumptalsvariansen leda till dåliga resultat. Vi hanterar detta genom att använda stora primtal som slumptalsfrön och genom att utföra identiska experiment med olika slumptalsfrön.
* Körande processer på distribuerade system/bakgrundsprocesser på lokala system. Om många processer kör samtidigt som vi utför experiment på t.ex. KTH’s Linux servrar så kan processen vi kör få låg prioritet. Resultaten kan således bli olika för identiska experiment. Felet minimeras genom att köra experimenten flera gånger för identiska in-parametrar.

## 4.3 Validering av experimentuppställningen

Experimentuppställningen valideras genom att algoritmernas teoretiska tids- och minneskomplexitet i bästa-, värsta- och normalfall jämförs med resultaten från experimenten.

## 4.4 Villkorlig förändring av experimentplanen

Om resultaten från experimenten inte stämmer överens med algoritmernas teoretiska tidskomplexitet måste experimenten ändras.

## 4.5 Hur induktiva respektive deduktiva metoder kan användas

Induktiva metoder kan användas för att dra slutsatser kring algoritmernas prestanda baserat på experimentell data. Deduktiva metoder kan användas för att dra slutsatser kring den teoretiska tids- och minneskomplexiteten hos algoritmerna.

# 5. När är målen uppfyllda? När är experimenten färdiga?

## 5.1 När är de identifierade målen uppfyllda

Resultatmålen är uppfyllda när vi har producerat en teknisk rapport och en experimentplanering som uppfyller kraven i kravanalysen samt att vi har gjort en utvärdering av de fyra algoritmerna baserat på insamlad experimentell data.

## 5.2 När är experimenten färdiga

Experimenten är färdiga när vi har samlat in tillräckligt med data för at utvärdera de fyra algoritmerna i bästa-, värsta- och normalfall.

# Appendix A: Kravanalys

I tabell A.1 finns en beskrivning av varje krav satt på experimentplaneringen, rapporten och utvärderingen av de fyra algoritmerna. I första kolumnen finns en numrering av de enskilda kraven för enklare referering. I andra kolumnen identifieras kravet. De enskilda kravens källor anges inte eftersom alla krav har samma källa. I tredje kolumnen beskrivs kriterierna för att kravet ska vara uppfyllt. I fjärde kolumnen beskriver vi till vilken grad kravet är uppfyllt. Eftersom alla krav är av typen ”måste” så beskrivs inte kravens typ i tabellen.

*tabell A.1: kravanalys som beskriver varje krav som har satts på labbuppgiften och arbetsmetodiken och hur de har uppfyllts.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Krav nummer** | **Namn/Beskrivning** | **Kriterier för att kravet ska vara uppfyllt** | **Om och till vilken grad kravet är uppfyllt** |
| 1 | Kravanalys | Alla enskilda krav på experimentplaneringen, experimenten och rapporten ska identifieras och beskrivas. Kravanalysen ska göras i form av en checklista i tabellform med fyra kolumner. Första kolumnen ska innehålla en numrering av de enskilda kraven. Andra kolumnen ska innehålla en benämning på varje enskilt krav samt källan till kravet. I tredje kolumnen ska tolkningen av kriterierna för varje enskilt krav beskrivas. I fjärde kolumnen ska en markering stå som indikerar om kravet är uppfyllt eller ej. Kravanalysen ska bifogas till experimentplaneringen. | Kravanalysen har skrivits enligt dem tolkade kraven. |
| 2 | Resultat och effektmål | Resultatmålen samt effektmålen ska identifieras och beskrivas. Med effektmål menas syftet med uppgiften. Kriterierna för när de enskilda resultatmålen är uppfyllda ska beskrivas. Beskrivningen av målen ska bifogas till experimentplaneringen. | Uppfyllt. |
| 3 | Förstudie | En förstudie ska göras där vi visar att vi har förstått uppgiften och systemet som ska studeras. Vi ska också beskriva vilka parametrar systemet beror av, vilka som kan varieras och hur de beror av varandra. Förstudien ska bifogas till experimentplaneringen. | Uppfyllt. |
| 4 | Experimentplan | En experimentplan ska skrivas där vi beskriver de enskilda experimenten som ska utföras. Vad som ska mätas och hur ska också beskrivas. Alla enskilda experiment ska motiveras. Kriterierna för när experimenten är färdiga ska beskrivas. Experimentplanen ska bifogas till experimentplaneringen. | Uppfyllt. |
| 5 | Krav på enskilda experiment | För experimenten ska det beskrivas vilka resurser som behövs, vilka osäkerhets/fel-källor som finns och hur experimentuppställningen valideras. För varje osäkerhets/fel-källa ska det beskrivas hur de kan påverka resultaten och hur osäkerheten kan minimeras. | Uppfyllt. |
| 6 | Insamling av data | I experimentplanen ska beskrivas vilken data som behöver samlas in, vilket format det att ha och hur vi planerar att analysera data. | Uppfyllt. |
| 7 | Villkorlig förändring av experimentplanen | I experimentplanen ska det beskrivas vad som skulle kunna leda till en förändring i experimentplanen. | Uppfyllt. |
| 8 | Slutsatser | I experimentplanen ska en diskussion återfinnas som diskuterar slutsatsernas karaktär d.v.s. om slutsatserna kommer kunna dras induktivt, deduktivt eller om de kommer att vara en kombination. När en deduktiv metod används måste slutsatserna följa från premisserna. När en induktiv metod används finns en viss osäkerhet. | Uppfyllt. |
| 9 | Studiens karaktär | I experimentplanen ska studiens karaktär beskrivas d.v.s. om studien är kvantitativ eller kvalitativ. I en kvantitativ studie kan svaren mätas och ges ett numeriskt värde. I en kvalitativ studie kan svaren enkelt ges numeriska värden. | Uppfyllt. |
| 10 | Utvärderingen | I uppgiften ska fyra olika implementationer av prioritetsköer baserade på länkade listor utvärderas:   * En enkellänkad lista där insättningar av nya element sker framifrån. * En dubbellänkad lista där insättningar av nya element sker från slutet av listan. * En dubbellänkad lista där insättning av ett nytt element sker framifrån om prioriteten på det nya elementet är högre än medelvärdet av prioriteterna på det första och sista elementet i listan. Annars sker insättning från slutet av listan. * En prioritetskö där prioriteterna är heltal i intervallet [0,40] där lågt numeriskt värde är hög prioritet. För varje prioritet skall finnas en kö. Varje delkö betjänas enligt FIFO.   Utvärderingen ska omfatta analytiska och exekveringsmässiga bästa-, värsta- och normalfallsprestanda.    Användningsfallen ska innefatta två fall:   * Prioritetsköerna används som schemaläggningsköer för processer med heltalsprioriteter i ett begränsat intervall [0, 40]. Processen som tas ut kommer antingen att läggas tillbaka, tas bort eller generera fler processer. Dem nya processerna har nya, högre tidsstämplar. * Prioritetsköerna används som händelselistor i en händelsestyrd simulering. Prioriteten modelleras som en tidsstämpel av datatypen double. Lågt numeriskt värde betyder hög prioritet. Simuleringen ska modelleras som att man tar ut den händelse som har lägst tidsstämpel T ur kön. Detta skall generera 0-N nya händelser med nya tidsstämplar T+increment() där increment() är ett icke-negativt stokastiskt genererat decimaltal. Om två händelser med identiska tidsstämplar genereras så ska de läggas i kön i den ordning de genererades.   Algoritmerna ska implementeras i programmeringsspråket C. | Ej uppfyllt. |
| 11 | Rapportens utformning | Resultatet av uppgiften ska vara en teknisk rapport. Rapporten skall omfatta 7-12 sidor exkluderat försättsblad, Abstract, sammanfattning, innehållsförteckning, referenser och bilagor. Rapporten kan skrivas i grupper om två och ska lämnas in individuellt i bilda. | Ej uppfyllt. |