

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII ȘTIINȚIFICE



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

DEPARTMENTUL CALCULATOARE

Titlul lucrării de disertație

Lucrare de disertație

Absolvent

Nume prenume student

Conducător

Titlul științific și numele cadrului diactic coordonator

Februarie/Iunie/Septembrie 2015



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

DEPARTAMENTUL CALCULATOARE

DECAN FACULTATE
Prof. dr. ing. Liviu MICLEA

DIRECTOR DEPARTAMENT
Prof. dr. ing. Rodica POTOLEA

Titlul lucrării de disertație

Lucrare de disertație

1. Absolvent: Nume prenume student
2. Conducător: Titlul științific și numele cadrului diactic coordonator
3. Conținutul lucrării: Pagina de prezentare, aprecierile coordonatorului, titlul capitolului 1, titlul capitolului 2, ..., titlul capitolului n, bibliografie, anexe, CD.
4. Locul documentării: UTCN, Cluj-Napoca
5. Consultanți: Donald Knuth, Leslie Lamport, others ...
6. Data emiterii temei:
7. Data predării:

Semnătură Conducător
Titlul științific și numele cadrului diactic coordonator

Semnătură Absolvent
Nume prenume student

Februarie/Iunie/Septembrie 2015



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

DEPARTMENTUL CALCULATOARE

Declarație pe proprie răspundere privind autenticitatea lucrării de licență

Subsemnatul *Nume prenume student*, legitimat cu *CI/BI* seria *XX* numărul *NN-NNNN*, CNP *LLLLLLLLLLLLL*, autorul lucrării *Titlul lucrării de disertație* elaborată în vederea susținerii examenului de finalizare a studiilor de masterat la Facultatea de Automatică și Calculatoare, Departamentul Calculatoare, Specializarea *SSSSSSSS* din cadrul Universității Tehnice din Cluj-Napoca, sesiunea *Februarie/Iunie/Septembrie* a anului universitar *20XX/20XX*, declar pe proprie răspundere, că această lucrare este rezultatul propriei mele activități intelectuale, pe baza cercetărilor mele și pe baza informațiilor obținute din surse care au fost citate în textul lucrării și în bibliografie.

Declar că această lucrare nu conține porțiuni plagiate, iar sursele bibliografice au fost folosite cu respectarea legislației române și a convențiilor internaționale privind drepturile de autor.

Declar, de asemenea, că această lucrare nu a mai fost prezentată în fața unei alte comisii de examen de licență sau disertație.

În cazul constatării ulterioare a unor declarații false, voi suporta sancțiunile administrative, respectiv, *anularea examenului de disertație*.

Cluj-Napoca
data

Semnătură
Absolvent

Rezumat

Descrierea sumară a lucrării, în câteva fraze. Un site foarte util ce conține exemple \LaTeX se găsește la <http://en.wikibooks.org/wiki/LaTeX>. Recomandăm crearea unui proiect în editoare Latex specializate (exemplu Kile pe Linux, TexnicCenter pe Windows) astfel încât compilarea/translationarea codului Latex în PDF să se facă din orice fișier editați la un moment dat (practic se va compila proiectul). De asemenea, dacă fișierul “thesis.bib” este inclus în proiect, există facilitatea de code-completion și la cite.

Cuprins

Listă de tabele	iii
Listă de figuri	v
1 Introducere	1
1.1 Context	1
1.2 Motivație	2
1.3 Structura lucrării	2
2 Obiective și specificații	3
2.1 Obiective	3
2.2 Specificații	3
2.2.1 Specificații funcționale	3
2.2.2 Specificații non-funcționale	4
3 Studiu bibliografic	5
3.1 Abordări similare	5
3.2 Tehnici/Tehnologii/Surse folosite	5
4 Fundamente teoretice	7
4.0.1 Identificarea de tipare secvențiale	7
5 Prezentarea contribuțiilor autorului	9
5.1 Precizări asupra conținutului și a modului de organizare	9
5.1.1 Dimensiune	9
5.2 Examples: lists, figures, tables, equations	9
6 Rezultate teoretice și experimentale	13
6.1 Teste de funcționalitate	13
6.2 Teste de performanță	13
7 Concluzii	15

A	Diverse anexe	17
B	Demonstrații matematice detaliate (dacă există)	19
C	Pseudo-cod sau cod (dacă există)	21
D	Articole publicate	23

Listă de tabele

5.1	Nonlinear Model Results	10
-----	-----------------------------------	----

Listă de figuri

5.1	Numele figurii	10
-----	--------------------------	----

Capitolul 1

Introducere

În cadrul acestui proiect se încearcă identificarea unei metode care rezolvă problema silabificării.

O astfel de metodă este de interes într-o serie de domenii conexe lingvisticii, spre exemplu în cadrul editoarelor de text sau pentru sinteza artificială a limbajului (audio).

Soluția propusă se folosește de metode de învățare supervizată bazându-se tipare secvențiale.

Ce se scrie aici:

- Contextul
- Conturarea/Descrierea domeniului exact al temei
- Se răspunde la întrebările: **ce** (s-a făcut)?, **de ce** (s-a făcut, adică motivația; ce se rezolvă, la ce este util, etc.)?, **cum** (s-a făcut, adică particularitățile abordării, prezentate sumar).
- Introducerea se termină cu o descriere a conținutului lucrării, de genul: Cap X descrie ..., Cap Y prezintă ...
- Introducerea reprezintă o sinteză a lucrării, din care cititorul trebuie să-și poată da bine seama dacă lucrarea prezintă sau nu interes pentru el.
- Se poate organiza pe subsecțiuni, dacă se dorește, după exemplul de mai jos, dar nu e obligatoriu asta, având în vedere dimensiunea mică
- reprezintă cca 5% din lucrare (nu mai mult de 2-4 pagini)

1.1 Context

Despre contextul în care este abordată și se aplică tema lucrării.

1.2 Motivație

De ce este utilă abordarea temei? Ce probleme rezolvă și ce rezultate poate aduce?

1.3 Structura lucrării

Capitolul 2 prezintă obiectivele Capitolul 3 descrie În capitolul 4 sunt prezentate

Capitolul 2

Obiective și specificații

Acest capitol conține descrierea detaliată a temei de cercetare propriu-zise, formulată exact, cu obiective clare și specificații, pe 2-3 pagini și eventuale figuri explicative. Titlul nu e neapărat impus și, de asemenea, capitolul poate fi inclus ca subcapitol în Capitolul 1, dacă se potrivește.

Reprezintă cca. 5–10% din lucrare.

2.1 Obiective

Obiectivele proiectului sunt lucrurile care se dorește a fi realizate, ca urmare a abordării temei lucrării de disertație. În general numărul de obiective este proporțional cu timpul de care dispunem. Exemple generice:

1. Analiza critică a soluțiilor existente pentru problema abordată și identificarea posibile limitări ale acestora.
2. Propunerea unor soluții la (o parte) din problemele identificate.
3. Implementarea unui/unor prototipuri de validare și testare a soluțiilor propuse
4. Identificarea unor teme de dezvoltare și cercetare ulterioare
5. ...

2.2 Specificații

2.2.1 Specificații funcționale

Soluția noastră:

- va face următoarele ...

- va oferi următoarea funcționalitate ...
- va fi bazată pe modelul ... (client-server)
- va fi implementată în C, Java etc.
- ...

2.2.2 Specificații non-funcționale

Soluția/Prototipul dezvoltat ar trebuie, de asemenea, să aibă următoarele caracteristici non-funcționale (exemple):

- să aibă următoarea performanță
- să fie ușor/intuitiv de utilizat
- să fie adoptată pe scară largă
- ...

Capitolul 3

Studiu bibliografic

Documentarea bibliografică are ca obiectiv fixarea referențialului în care se situează tema, prezentarea surselor bibliografice utilizate și a cercetărilor similare și raportarea abordării din lucrare la acestea.

Referințele bibliografice se vor face pentru fiecare carte, articol sau material folosit pentru elaborarea lucrării de licență.

Reprezintă cca. 10–15% din lucrare.

3.1 Abordări similare

Comparați abordarea voastră cu cele ale altor soluții: ce e asemănător, ce e diferit (și, de preferat, mai bun).

Citarea referințelor se face ca în exemplele ?? din Bibliografie. Vezi citările următoare.

În articolul [?] autorul descrie configurația tehnică a unei "honeynet" și prezintă câteva atacuri de actualitate asupra honeynet, precum și o serie de recomandări pentru securizarea sistemelor conectate la rețele de calculatoare.

În articolul on-line [?] găsim detalii interesante despre

3.2 Tehnici/Tehnologii/Surse folosite

Sursele de documentare referitoare la metodele, tehnologiile, ideile folosite.

Capitolul 4

Fundamente teoretice

4.0.1 Identificarea de tipare secventiale

Aici se descriu pe scurt aspecte teoretice pe care se bazează lucrarea. Conținutul acestui capitol trebuie gândit pentru un cititor care nu e specializat pe domeniul temei și nu cunoaște chestiunile de bază despre subiect. Pentru un cititor specializat, capitolul poate să stabilească un limbaj comun, relativ la termenii care pot fi interpretați diferit.

Acest capitol nu trebuie gândit și scris nici ca un copy-paste din alte surse, nici ca zona de reglaj a numărului de pagini ale lucrării. Deși va conține chestiuni pe care le-ați studiat și voi și pe care v-ați bazat, el trebuie să fie o compilare a surselor folosite, care să aibă sens și relevanță pentru lucrarea voastră. Trebuie să fie o descriere coerentă și logică a unor aspecte care ușurează sau fac posibilă înțelegerea părților următoare ale lucrării. Nu trebuie intrat însă prea mult în detalii, ci spuse doar chestiunile esențiale.

Dacă preluați text, figuri, tabela etc. din sursele de documentare, acestea din urmă trebuie indicate explicit.

Reprezintă cca. 10% din lucrare.

Capitolul 5

Prezentarea contribuțiilor autorului

5.1 Precizări asupra conținutului și a modului de organizare

Titlul acestui capitol nu este unul impus și nici nu corespunde neapărat unui singur capitol. Titlul indică mai degrabă o parte (importantă și centrală, de altfel) a lucrării, în care se prezintă ceea ce s-a realizat efectiv: contribuțiile autorului. Organizarea acestei părți este dependentă și specifică fiecărei lucrări în parte și este stabilită de către fiecare autor după cum i se pare mai potrivit pentru tema lui. Ea poate cuprinde prezentarea unor concepte teoretice (unelte sau tehnici matematice folosite în lucrare, prezentarea sau introducerea unor concepte teoretice etc.), o analiză a diferitelor metode/algoritmi/tehnologii etc. luate în considerare sau dezvoltate de către autor, o prezentare a unui design (mai mult sau mai puțin detaliat) sau chiar detalii a unei eventuale implementări/prototip, dacă e cazul.

Trebuie remarcat însă faptul că această parte reprezintă contribuția personală a autorului, chiar dacă ea constă de exemplu doar dintr-o analiză comparativă a unor metode/algoritmi, și în nici un caz ea nu poate fi sinteza unor texte preluate din alte surse. Prin urmare, orice informații sunt prezentate aici, ele trebuie să corespundă cel puțin unei interpretări/analize critice personale a autorului, dacă nu chiar unor idei originale ale acestuia.

5.1.1 Dimensiune

Împreună cu capitolul (partea) următor reprezintă cca. 70% din lucrare.

5.2 Examples: lists, figures, tables, equations

Așa arată o listă de elemente nenumerotate:

Case	Method#1	Method#2	Method#3
1	50	837	970
2	47	877	230
3	31	25	415

Tabela 5.1: Nonlinear Model Results

- element 1
- element 2
- ...

Așa arată o listă de elemente numerotare:

- element 1
- element 2
- ...

Așa arată o listă în text: (1) element 1, (2) element 2, (3) ...

Atenție: orice tabel, figura sau ecuație (formulă) trebuie referite *explicit* în text explicit (de genul: în Figura X este ulustrat ..., în Tabelul Y se poate vedea ...), pentru că Latex le poate plasa chiar și pe altă pagină decât acolo unde vrem noi să ne referim la ele. Vedeți exemple de mai jos!

Tabelul 5.1 ilustrează un exemplu de tabel. Un editor on-line de tabele poate fi găsit la <http://www.tablesgenerator.com/>.

În Figura 5.1

Formula (5.1) arată modul de calcul al lui Δ :

$$\Delta = \sum_{i=1}^N w_i (x_i - \bar{x})^2. \quad (5.1)$$

Algoritmul 1 este un exemplu de descriere pseudo-cod a unui algoritm, preluat de la <http://en.wikibooks.org/wiki/LaTeX>. El utilizează pachetul *algorithm2e*. Alternativ, puteți utiliza pachetele *algorithmic* sau *program*.

Data: this text

Result: how to write algorithm with L^AT_EX2e initialization;

```
while not at end of this document do
|   read current;
|   if understand then
|   |   go to next section;
|   |   current section becomes this one;
|   else
|   |   go back to the beginning of current section;
|   end
end
```

Algorithm 1: How to write algorithms

Capitolul 6

Rezultate teoretice și experimentale

Împreună cu partea de prezentare a proiectului, trebuie să reprezinte aproximativ 70% din lucrare.

Aici sunt prezentate metodele de validare a soluțiilor/sistemului descris în capitolele anterioare, scenariile de testare a corectitudinii funcționale, a utilizabilității, performanței etc.

Rezultatele testelor experimentale necesită, în general interpretări (dacă rezultatele obținute corespund așteptărilor, intuițiilor cititorului, de ce apar variații/exceptii etc.) și comparații cu rezultatele altor metode similare.

Sistemele de testare și testele propriu-zise trebuie descrise detaliat astfel încât să poată fi reproduse și de alții care poate vor să-și compare soluțiile lor cu a voastră (eventual, codul testelor poate fi pus în anexe). Dacă se poate alegeți pentru evaluarea sistemului vostru benchmark-uri (pachete de testare) dedicate, astfel încât comparația cu alte sisteme să poată fi făcută mai ușor. În plus, astfel de teste sunt mult mai complete și mai realiste decât cele dezvoltate de voi. Oricum, încercați ca testele efectuate să nu fie triviale, ci să acopere scenarii cât mai reale, mai complexe și mai relevante ale funcționării sistemului vostru.

6.1 Teste de funcționalitate

6.2 Teste de performanță

Capitolul 7

Conluzii

Cuprinde:

- un rezumat al contribuțiilor aduse: ce s-a realizat, relativ la ce s-a propus, în ce constă experiența acumulată, care au fost punctele dificile atinse și rezolvată, recomandări pentru alții care abordează tema, la ce este bun ce s-a obținut etc.
- a analiză critică a rezultatelor obținute: avantaje, dezavantaje, limitări
- o descriere a posibilelor dezvoltări și îmbunătățiri ulterioare

Poate fi organizat pe secțiuni, dacă se dorește.
Se întinde pe aproximativ 1-2 pagini.

Anexa A

Diverse anexe

Anexa B

Demonstrații matematice detaliate (dacă există)

Anexa C

Pseudo-cod sau cod (dacă există)

```
/** Maps are easy to use in Scala. */
object Maps {
  val colors = Map("red" -> 0xFF0000,
                   "turquoise" -> 0x00FFFF,
                   "black" -> 0x000000,
                   "orange" -> 0xFF8040,
                   "brown" -> 0x804000)

  def main(args: Array[String]) {
    for (name <- args) println(
      colors.get(name) match {
        case Some(code) =>
          name + " has code: " + code
        case None =>
          "Unknown color: " + name
      }
    )
  }
}
```


Anexa D

Articole publicate