FORMAL DERIVATION OF LAMPORT'S BAKERY ALGORITHM IN MULTIPROGRAMMING

A THESIS

By

RETZI YOSIA LEWU
00013053

INFORMATICS ENGINEERING DEPARTMENT
FACULTY OF ENGINEERING
DE LA SALLE CATHOLIC UNIVERSITY
MANADO

2004

FORMAL DERIVATION OF LAMPORT'S BAKERY ALGORITHM IN MULTIPROGRAMMING

A THESIS

Presented as a Partial Fulfillment of the Requirements
in obtaining *Sarjana Teknik* Degree in Informatics Engineering Department

By

RETZI YOSIA LEWU
00013053

INFORMATICS ENGINEERING DEPARTMENT
FACULTY OF ENGINEERING
DE LA SALLE CATHOLIC UNIVERSITY
MANADO

2004

DE LA SALLE CATHOLIC UNIVERSITY
MANADO - INDONESIA

Name	:	Retzi Yosia Lewu
NIM	:	00013053
Faculty	:	Engineering
Department	:	Informatics Engineering
Title of Thesis	:	Formal Derivation of Lamport's Bakery Algorithm in Multiprogramming
Supervisor	:	Dr. Benny Pinontoan, M.Sc.

Approved,
Manado, 16 September 2004

Supervisor,

Dr. Benny Pinontoan, M.Sc.

Acknowledged,

Dean,	Head of Department,
Ir. Noldi Watuna, MM.	Ir. Simon Patabang, MT.

ABSTRACT

Algorithm is a step-by-step procedure to solve certain problem. It is later transformed into another form that is understandable for computer, called program. In designing an algorithm there are two things needed. They are efficiency and correctness. Program correctness is important since a program will be useless if it cannot perform an expected output. On the other side, efficiency is also important. In sequential programs, the statement is executed one after another. Meanwhile, there are some sections of that program that can be synchronized so that they can be executed simultaneously. Nevertheless, the synchronization process needs some rules or protocols so that they will do exactly the same as they were put sequentially. The big advantage here is the efficiency and it is the main issue of multiprogramming. The synchronized programs are called a multiprogram.

A problem may occur when those program are being synchronized. For example, in a problem named mutual Exclusion Problem, there are two component trying to enter critical section. A solution of this problem must ensure that the mutual exclusion property is satisfied and guarantee that there is no deadlock.

The subject concerned in this thesis, shows how the derivation of Lamport's Bakery Algorithm to solve the mutual exclusion problem is made simultaneously with its proof of correctness, both locally and globally. This work is done through the approximation method by using the Gries-Owicki Theorem and some logic and calculus. The approximation is done repeatedly to find a final solution that satisfies the given specification.

Keywords: **efficiency, correctness, sequential program, synchronization, multiprogramming, multiprogram, The Gries-Owicki Theorem, local correctness, global correctness, Lamport's Bakery Algorithm, Mutual Exclusion, deadlock.**

ABSTRAK

Algoritma adalah langkah-langkah yang harus diambil dalam memecahkan suatu masalah. Algoritma ini kemudian diterjemahkan dalam suatu bentuk yang dapat dipahami oleh komputer, yang disebut program. Dalam mendisain sebuah alagoritma, diperlukan dua hal penting yaitu efisiensi dan kebenaran. Kebenaran sebuah program sangatlah penting mengingat sebuah program disebut *useless* jika tidak dapat memberikan output yang sesuai dengan yang diharapkan. Di lain pihak, efisiensi jugalah penting. Dalam program-program sekuensial, *statements* dieksekusi satu persatu. Padahal, ada beberapa bagian yang dapat disinkronisasi sehingga mereka dapat dieksekusi secara bersama-sama. Namun demikian, proses sinkronisasi ini membutuhkan beberapa aturan maupun protokol sehingga saat program dieksekusi, mereka akan melakukan hal yang sama dengan saat dieksekusi secara sekuensial. Manfaat terbesar yang dapat diperoleh adalah dalam hal efisiensi dan inilah isu terpenting dalam *multiprogramming*. Program-program yang telah disinkronisasi disebut multiprogram.

Namun, masalah dapat terjadi saat proses sinkronisasi itu dilakukan. Contohnya, dalam permasalahan yang lebih dikenal dengan *Mutual* Exclusion, ada dua komponen program yang mencoba untuk memasuki *critical section*-nya. Solusi dari permasalahan ini harus memastikan bahwa properti *mutual* exclusion terpenuhi dan menjamin bahwa tidak ada *deadlock*.

Pokok tulisan dalan Tugas Akhir ini menyangkut proses derivasi untuk *Lamport's Bakery Algorithm* untuk mengatasi masalah *mutual exclusion*. Proses derivasi dilakukan secara bersamaan dengan pembuktian kebenarannya baik secara lokal maupun global lewat metode aproksimasi dengan menggunakan teorema *Gries-Owicki*. Aproksimasi dilakukan secara berulang-ulang sehingga diperoleh hasil yang sesuai dengan spesifikasi yang telah diberikan.

Kata kunci: efficiency, correctness, sequential program, synchronization, multiprogramming, multiprogram, The Gries-Owicki Theorem, local correctness, global correctness, Lamport's Bakery Algorithm, Mutual Exclusion, deadlock.

ACKNOWLEDGEMENTS

Glory to GOD for His love and mercy, for guiding me during this research, for being with me always, and for being the most perfect person I ever had.

I would like to thank to my thesis supervisor Dr. Benny Pinontoan M.Sc.. He guided and supported my research with great care and patience. He taught me how to do the research, and did it well. Without his help, I would not have pulled into the topic.

I would also like to give a very special thanks and honor to the other examiners, Ir. Armein Langi, M.Sc.,Ph.D. and Dr. Rajesri Govindaraju, M.Sc., from Bandung Institute of Technology, for a number of critical comments and valuable suggestions during the examination. Your input is a great contribution and a new knowledge for me.

To Ir. Noldi Watuna, M.M , the Dean of Faculty of Engineering, thank you very much for your help, support and a chance to choose the thesis topic and also to put me in a great group. To Ir. Simon Patabang, M.T, the Head of Informatics Engineering Department (also as my Academic Advisor), thank you for the book and also for all your help during the research. To Computer Laboratories Officers, thank you for the chances to use the labs and also the Internet service.

To all my friends in the thesis group (under supervision of Dr Benny Pinontoan M.Sc.), Lanny Sitanayah (thank you for the journal you've got for me), Julia Dama (thank you for all your suggestions), Wenny Pramanto, Meidy Neghe, Lisa Wong and Ruland Rantung, thank you for being together during the research. I also want to thank to my thesis group mates from Computer Science Department, Yuniarti Halim, Yeyen Thadete and Inggrid Ulaan for their help.

To Mr. Chris Kahuweka, thank you for your time and willingness to check the spelling and grammar of this thesis. Also, I want to thank to Anto DP for his valuable contribution of my presentation materials and also for the printer he has lent me.

To my beloved family at home, Mom, Dad, Christie and my little brother Toar, thank you very much for your support and prayer.

In particular I also want to thank to my dearest friends: Ivonne Ombuh, Paula Kokoy, Conny Hongo, Dewi Angelina, Sandy Pesik and Ivanna Monintja. Without their love and support I would not finish this writing. I also want to thank to everyone whose support and prayer will always be with me.

Throughout the research I expect to offer a new point of view in multiprogramming. But I do realize there are still many lacks. That's why your participation here by some critics, suggestions and also comments for further improvement of this thesis, will be highly appreciated.

"..and that not of yourselves: it is the gift of GOD"

(Ephesians 2: 8b)

CONTENTS

LIST OF SYMBOLS

< Less than 21

≠ Not equal 24

{P} S {R} Hoare Triple: on precondition P, execution of S ensures post condition R 13

if…fi Alternative Construct or Conditional Rules 15

do…od Repetitive Construct or While Rule 15

wp Weakest Precondition 16

wlp Weakest Liberal Precondition 17

|| Multiprogram's notation, shows synchronization among program components 13

P||Q Multiprogramming of Programs P and Q 13

|[S]|* While true do S (S is a statement) 23

L Local Correctness 24

G Global Correctness 24

{…}P||{…}Q Assertions on both component program P and Q 24

Comp Component of multiprogram 24

Pre Precondition 24

Inv Invariant of a Multiprogram 27

cs Critical Section 31

ncs Non Critical Section 31

:: As 32

max maximum function from where a component choosing the greater number 37

LIST OF TABLES