

엡저 위베 다익스트라  
(Edsger Wybe Dijkstra,  
1930년 5월 11일 — 2002년 8월 6일)

김도형

성신여자대학교 컴퓨터정보학부

Do-Hyung Kim

School of CSE, Sungshin Women's University

E-mail: [dkim@sungshin.ac.kr](mailto:dkim@sungshin.ac.kr); URL: <http://cs.sungshin.ac.kr/~dkim>



2002년 8월 6일, 미국 텍사스 주립대학교 오스틴 분교 컴퓨터학과 및 수학과에 명예교수로 있던 엡저 위베 다익스트라가 네덜란드 뉴넨에 있는 그의 자택에서 유명을 달리하였습니다. 컴퓨터 분야의 진정한 거인(巨人) 중 한 명인 다익스트라의 별세를 애석해 마지않으며, 고인의 명복을 비는 바입니다.

필자는 수많은 컴퓨터 과학자들 중 다익스트라를 가장 좋아하면서 그의 사고(思考)와 글 쓰는 방식을 제가 추구해야 할 하나의 전범(典範)으로 생각해 왔었고, 아직까지는 기회가 없었지만 인젠가는 다익스트라를 직접 만나서 그의 명저(名著)인 '프로그래밍의 수련'을 번역하면서(아직 반의 반도 마치지 못 했지만) 느낀 어려움과 미진함을 이야기 할 수 있기를 마음 속으로 간절히 바래왔기에, 그의 부고에 느끼는 아쉬움은 참으로 큼니다. 다시 한번 더 고인의

명복을 빕니다.

다익스트라의 연구와 관심은 대단히 폭넓게 펼쳐져 있지만, 컴퓨터 과학의 세부 분야에서 주된 몇 가지를 꼽자면 아무래도 알고리즘과 프로그래밍 언어, 소프트웨어 공학 그리고 교육을 들어야 할 것 같습니다. 이런 점에서 우리 프로그래밍언어연구회가 공식 기관지를 통해 그에 대한 애도와 경의를 표하는 것은 당연하다고 생각되어, 그를 돌아보는 웹 상의 글과 개인적 소회(所懷)를 준비하게 되었습니다.

텍사스 주립대학교 컴퓨터학과 웹 사이트의 공식 부고(訃告)

(<http://www.cs.utexas.edu/users/EWD/obituary.html>)

컴퓨터 과학과 산업 분야의 유명한 선구자(先驅者)인 엡저 위베 다익스트라 교수가 암(癌)과의 오랜 투병 끝에 네덜란드 뉴넨에 있는 그의 자택에서 2002년 8월 6일 타계하였습니다.

다익스트라는 1930년 네덜란드 로테르담에서 화학자 아버지와 수학자 어머니의 아들로 태어났습니다. 그는 로테르담에 있는 에라스무스 김나지움을 졸업한 뒤 라이덴 대학교에서 수학과 이론물리학 학위를, 그리고 암스테르담 대학교에서 컴퓨터 과학으로 박사 학위를 받았습니다. 그는 1952년부터 1962년까지 암스테르담에 있는 수리과학연구소(Mathematisch Centrum)에서 프로그래머로 일했으며; 1962년부터 1984년까지 아인트호벤 공과대학교의 수학과 교수이면서; 1973년부터 1984년까지 버로우즈 사의 연구 고문(顧問)이었습니다. 그는 1984년부터 1999년까지 텍사스 주립대학교 오스틴 분교의 컴퓨터과학과에서 슬림버거 세기(世紀) 석좌교수(碩座教授)를 역임하였고, 1999년 명예교수로 퇴직하였습니다.

다익스트라의 유족으로는 40년 넘게 함께 살아온 아내 마리아(리아) C. 다익스트라 디베츠, 세 명의 자식 마르쿠스 J., 뫼케 E., 컴퓨터 과학자인 룯거 M. 다익스트라와 두 명의 손자들이 있습니다.

다익스트라는 종종 컴퓨터 분야의 노벨상으로 간주하는 미국컴퓨터협회(ACM) 튜링상의 1972년도 수상자입니다. 그는 네덜란드 왕립한림원(王立翰林院)과 미국 학술원의 회원이었고, 영국 컴퓨터학회의 석좌(碩座) 펠로우였습니다. 그는 1974년도 전미정보처리단체연맹(AFIPS)의 해리 굿 상, 1982년도 세계전기전자공학자협회(IEEE)의 컴퓨터 선구자 상, 그리고 컴퓨터 과학 교육에 대한 최고 공헌으로 1989년도 미국컴퓨터협회 컴퓨터과학교육분과회(SIGCSE) 상을 수상하였습니다. 아테네 경제대학교는 2001년 그에게 명예박사 학위를 수여하였습니다. 2002년 일본의 C&C 재단은 “소프트웨어 기초 이론, 알고리즘 이론, 구조적 프로그래밍, 그리고 세마포 등에 관한 창의적(創意的) 연구를 통해 컴퓨터 소프트웨어를 위한 과학적 토대의 확립에 대한 다대(多大)한 공로”로 다익스트라를 표창하였습니다.

다익스트라는 수학적 논리가 적절한 컴퓨터 프로그램을 구성하기 위한 초석(礎石)이며 또 그레야만 한다는 통찰(洞察)과 수학적 방법론에 대한 기여로 유명합니다. 명시적으로 동기화 되는 순차 프로세스들로서 운영체제를 구성한다는 착상이나 컴퓨터 프로그램의 정형적 개발, 그리고 비결정성의 체계적 제어를 위한 지적 토대 등은 그의 공헌입니다. 놀랄 정도로 효율적인 최단(最短) 경로를 구하는 알고리즘과 최초의 Algol 60 컴파일러를 설계하고 작성한 것으로도 널리 알려져 있습니다. 과거 그는 프로그래밍에서 분기문(分岐文)을 제거하는 데 앞장선 것으로도 유명합니다.



다익스트라는 비상한 글 솜씨의 소유자이기도 합니다. 1,300쪽이 넘는 그의 저작 전체는 디지털로 스캔되어 <http://www.cs.utexas.edu/users/EWD>에서 볼 수 있습니다. 그는 또한 오랫동안 정기적으로 수많은 친구 및 동료들과 전자우편이 아닌 전통적인 방식으로 편지를 주고받았습니다. 그는 학문적 저작물과 편지를 작성하는 데 컴퓨터보다 만년필을 고집스럽게 선호(選好)하였습니다.

다익스트라는 기지(機智)와 달변(達辯), 그리고 어휘를 사용하는 그만의 방식으로 유명하였습니다: 예를 들면, “컴퓨터가 사고(思考)를 할 수 있는냐는 질문은 잠수함이 수영(水泳)을 할 수 있는냐는 질문과 같은 것이다”는 그의 언급<sup>1)</sup>이나; 연구 주제 선택 방법에 대해 묻는

1) 강(強) 인공지능주의자가 아주 좋아할 표현입니다. 또한 범인(凡人)들이 간과(看過)하고 오해(誤解)하기 쉬운 부분을 직격(直擊)하여 새롭게 각성(覺醒)시키는 정말 멋들어진 표현입니다.

유명한 연구자에게 “당신만이 할 수 있는 것만 연구하라”는 그의 조언에서나; “도구로서의 능력 면에서 볼 때 컴퓨터는 우리 문화의 표피(表皮)에 이는 잔물결에 불과할 것이다. 그러나 지적 과제로서의 능력 면에서 볼 때 컴퓨터는 인류의 문화사(文化史)에서 그 유례(類例)가 없는 것이다”는 그의 튜링상 수상 강연에서의 논평<sup>2)</sup>에서와 같은 식입니다.

다익스트라는 구조적 프로그래밍, 관심사(關心事)의 분리, 동기화, 치명적(致命的)인 밀착(密着), 식사하는 철학자(哲學者)들, 최약(最弱) 사전조건, 가드 명령어, 기적(奇蹟) 배제, 그리고 컴퓨터 프로세스들을 제어하기 위한 그 유명한 ‘세마포’와 같은 수많은 개념과 용어들로 컴퓨터 과학의 언어를 풍요(豐饒)롭게 만들었습니다. 옥스퍼드 영어사전은 컴퓨터와 관련된 문맥에서 다익스트라가 사용한 ‘백터’와 ‘스택’을 인용하고 있습니다.

다익스트라는 자신의 비젠도르퍼 피아노로 친구들을 위해 모차르트 연주하는 일을 즐겼습니다. 그와 그의 아내는 여행 기계<sup>3)</sup>라고 이름 붙인 자신들의 폴크스바겐 버스를 타고 미국의 국립공원과 주립공원을 돌아다니는 것을 좋아하였습니다. 바로 이 버스 안에서 그는 많은 학술 논문을 작성하였습니다.

자신의 과학자 경력의 처음부터 끝까지, 다익스트라는 상업적이거나 관료적 혹은 정치적 고려에 의해 더럽혀지지 않은 과학적 엄정(嚴整)함이라는 지고(至高)한 학문적 이상(理想)을

---

물론 다익스트라가 이런 말을 했다고 해서 그가 지능의 징표(徵標)를 확인하는 튜링 테스트와 같은 것에 전적으로 동의하는, 인공지능에 대한 완전한 행동주의자(行動主義者, behaviorist) 또는 기능주의자(機能主義者, functionalist)의 입장을 취한다고는 전혀 생각되지 않습니다. 다만 경박한 반(反) 인공지능주의자들의 비판에 대한 제지 있는 반박일 것입니다.

2) 이 논평은 다익스트라의 튜링상 강연에서 마지막 결론(結論) 부분에 나오는 것입니다. 이 부고에는 튜링상 강연에서 해당되는 부분을 그대로 옮기지는 않고 문장이 약간 재구성되어 있습니다. 물론 뜻은 그대로입니다. 단순히 이 부분을 번역하는 것만으로는 원래의 뜻이 제대로 전달될 것 같지 않아서 약간 부연(敷衍)을 하고자 합니다. (다익스트라의 튜링상 강연을 이미 읽으신 분들께는 사족이 될 것 같아 죄송합니다.) 여기서 다익스트라가 컴퓨터의 ‘도구로서의 능력’을 과소평가하고 있다고 판단하는 일은 성급한 것입니다. 어쩌면 다익스트라는 튜링상 강연을 할 당시에는 컴퓨터라는 도구가 인류의 문화와 문명에 막대(莫大)한 영향을 끼친 여러 다른 도구들(예컨대, 언어나 불, 문자, 종이, 바퀴, 화폐 등)에 비해 아직 미약하다고 생각했을 수도 있습니다. 그러나 그것보다는 컴퓨터의 ‘지적 과제로서의 능력’이 워낙 ‘도구로서의 능력’에 비해 잠재력이 크기에 그 점을 상대적으로 강조하기 위해 이런 표현을 사용하였다고 보는 것이 옳을 것입니다. 컴퓨터의 ‘지적 과제로서의 능력’이란 컴퓨터를 사용하기 위해 인간이 하는 일, 즉 프로그래밍과 그 결과물인 프로그램이 도대체 무엇이며, 어떻게 해야 올바르게 하는 것인지 등에 대해 아직 우리가 너무나 모르고 있다는 것을 의미합니다. 인류가 만든 어떤 개념이나 물건도 그 본질에 대한 기본적인 이해가 천박(淺薄)하기로는 컴퓨터에 비할 것이 없다는 것입니다. 그 가장 기본적인 사용 방식인 프로그래밍이라는 작업을 어떻게 하는 것이 옳은지에 대해서조차 대부분이 동의하는 방법론(方法論) 하나 없는 실정입니다. 비록 다른 도구가 컴퓨터보다 훨씬 오랜 역사(歷史)를 가지면서 이미 인류의 문화와 문명에 끼친 영향력이 막대하나, 그러한 도구들의 본질에 대해 우리가 컴퓨터처럼 모르는 것은 없습니다(그런 도구들이 처음 등장했을 때조차도 말이죠). 사실 생각해 보면 이러한 것은 당연한 일입니다. 인류가 지금까지 만든 도구의 대부분은(사실 컴퓨터를 제외하면 전부라고도 말할 수 있지만) 인간의 육체적인 한계를 보조하고 확대하는 것인데 반하여, 컴퓨터는 인간의 정신적인 한계를 보조하고 확대하는 것이 그 궁극적 목표이기 때문입니다. 성격이 완전히 다른 도구라는 뜻이지요. 이제 인류가 컴퓨터란 도구의 본질을 이해하게 되면 그것은 곧 문제 해결의 본질, 사고(思考)의 본질, 지능(知能)의 본질에 접근하는 일이 될 것입니다. 이러한 차원에서 컴퓨터의 ‘지적 과제로서의 능력’은 그 잠재력(潛在力)이 무한(無限)하다고 봅니다. 바로 이런 상대적으로 보다 중요한 측면을 강조하는 다익스트라다운 어법(語法)입니다.

3) 이 ‘여행 기계’는 원문을 그대로 적거나 적어도 발음나는 대로 적어야 그 맛을 살릴 수 있습니다. 원문에서 이것은 ‘Touring Machine’입니다. 튜링 기계(Turing Machine)의 언어 유희(遊戯)이지요. 다익스트라다운 명명(命名)입니다.

명료(明瞭)하게 말하고 추구하였습니다. 단순성, 아름다움, 그리고 달변은 그의 보증수표와 같았고, 프로그래밍과 수학에 있어서의 우미(優美)함에 대한 타협하지 않는 그의 주장은 수많은 사람들에게 영감(靈感)이 되었습니다. 그는 가장 높은 잣대를 가지고 스스로의 작업을 평가하였으며, 많은 동료들이 그와 같이 하도록 끊임없이 목표를 설정하였습니다. 다른 사람들을 위해 그는 기꺼이 소크라테스의 역할을 떠맡았는데, 그것은 잠시 유행하는 생각에 내재하는 본질적 오류와 시류(時流)에 영합하는 타협의 위험성에 관해 말함으로써 반복하여 자신의 모국과 거주국을 자극하는 사회의 등애와 같은 것이었습니다. 소크라테스와 마찬가지로, 그의 가장 큰 유산을 받은 사람들은 작은 집단의 토론에서 그와 함께 한 사람들이나 미완성의 개념과 새로운 발견에 관해 그와 기술적인 편지를 주고받은 사람들입니다. 특히 '화요일 오후 구락부(俱樂部)'로 알려진, 아인트호벤과 오스틴에서의 독회(讀會)에 참석한 사람들은 특전(特典)을 누린 셈입니다.

다익스트라의 작고에 이르러, 파이돈(Phaedo)<sup>4</sup>에서 소크라테스에 대해 언급한 것을 상기하고자 합니다: “우리가 알고 있는 그와 동시대의 사람들 중, 그가 가장 현명하고 공평한 최고의 인물이었다고 우리는 진심으로 말할 수 있을 것입니다.”

#### 텍사스 주립대학교 컴퓨터과학과 학과장의 추도사(追悼辭) (<http://www.cs.utexas.edu/users/EWD/JSMremarks.html>)

영어로 말하는 것을 용서해 주시기 바랍니다.

저는 텍사스 주립대학교 컴퓨터과학과 학과장이며, 다익스트라는 우리 학과에서 슬림버거 세기 석좌교수를 역임하였습니다.

다익스트라는 근 20년 간 우리들의 가족(家族)이었습니다. 그는 우리들 중 다수에게 아버지와 같은 존재였지요. 그러나 그는 또한 동료이며 친구이기도 하였습니다.

그를 지나치며 아는 정도인 사람들은 종종 저에게, “동료 교수로 다익스트라가 있는 것은 틀림없이 끔찍할 것 같네요”라고 말하곤 하였습니다. 그의 거침없는 태도가 불화(不和)를 일으키지나 않을까 그들은 염려하였던 것이지요. 그러나 다익스트라의 경우는 그렇지 않았습니다.

다익스트라는 기막힌 동료였습니다. 그는 학생들을 최상(最上)의 방법과 열정(熱情)으로 가르쳤습니다. 그는 교수회의에 단 한 가지 의제(議題)만 들고 왔습니다: 그것은 “어떻게 하면 우리가 나아질 수 있을까? 어떻게 하면 보다 나은 과학자가 될 수 있을까? 보다 나은 학자는? 보다 나은 선생은?”라는 것이었지요. 그는 자신의 입장을 툭 털어놓고 그것을 논의하였습니다. 그런 뒤 자신의 한 표(票)를 행사하였고, 그것이 다였습니다.

그는 결코 남몰래 학과장실로 와서 자기가 옛저 다익스트라이니까 우리가 그의 방식대로 일을 처리해야 한다고 주장하지 않았습니다.

저는 다익스트라같은 교수가 학과에 열 명 정도 있었으면 하고 바랬습니다. 아, 그러나 안타깝게도 다익스트라는 딱 한 명밖에 없었지요.

그는 종종 카우보이 모자와 티셔츠 차림으로 출근하였습니다. 허리 위로는 그는 저보다

4) 플라톤(Plato)의 유명한 저서입니다. 소크라테스가 독미나리(hemlock)로 만든 독배(毒杯)를 받고 죽는 장면을 배경으로 자살, 이데아, 영혼의 불멸성(不滅性) 등 여러 철학적 문제를 토론하고 있습니다.

더 텍사스 사람처럼 보였습니다. 그런데 그는 또한 반바지를 입고 샌들을 신고 작은 가방을 가지고 다녔습니다. 이것 때문에 텍사스 사람 이미지가 깨졌지요. 그는 다른 사람과 달랐고 우리는 그것이 더 좋았습니다.

그는 마치 어둠 속에서 등불을 가진 사람 같았지요. 그는 토의(討議)하는 거의 모든 문제에 대해 빛을 비추었습니다. 그와 만날 때마다 저는 뭔가를 배웠습니다.

그래요, 그는 위대한 컴퓨터 과학자였습니다. 그는 우리가 사용하는 많은 용어를 정의하였습니다. 그는 많은 중요한 개념을 창안(創案)하였습니다. 그는 과학의 거인들 중 한 명으로 기억될 것입니다. 그러나 텍사스 주립대학교 오스틴分校의 우리들은 그를 단순한 존경(尊敬)과 경외(敬畏)를 넘어서 기억할 것입니다. 우리는 그를 사랑과 애모(愛慕)를 담아 기억할 것입니다.

감사합니다.

2002년 8월 10일  
네덜란드 히즈에서  
J. 스트로더 무어

## 왜 다익스트라인가?

다익스트라에 대한 개인적인 생각과 느낌을 간단히 적고 싶습니다.

많은 분들이 다익스트라를 처음 접하는 시점이 학부 때 ‘자료구조’나 ‘알고리즘’, ‘운영체제’ 또는 ‘소프트웨어 공학’ 과목을 배울 때일 것입니다. 저 역시 예외가 아니었지요. 다익스트라의 최단 경로 알고리즘을 배우면서 “참 명쾌하면서도 깔끔하다!”고 생각하였습니다. 그러면서 그러한 알고리즘을 작성한 사람의 논리적이고 엄격(嚴格)한 성격을 알 듯한 느낌도 받았습니다. 그 외 여러 부분에서 그의 이름을 만나면서 다익스트라가 대단히 정력적인 연구자이기도 하다는 것을 알았습니다(당시 학부 학생으로서 부끄럽게도 튜링상도 모르던 시절이니까 이런 주제넘은 생각을 하였던 것이지요). 또한 ‘식사하는 철학자’를 배우면서 기발(奇拔)하고 유머러스한 다익스트라의 또 다른 면을 발견하였습니다. 매우 매력적(魅力的)인 사람이라고 생각하였습니다.

그러나 다익스트라를 결정적으로 좋아하게 된 것은 석사 과정 때 그의 저작(著作)인 ‘프로그래밍의 수련(A Discipline of Programming)’을 접한 것이 계기가 되었습니다. 지도교수님이셨던 KAIST의 최광무 교수님이 추천해 주셨는데, 추천의 변(辯)은 “수필(隨筆) 읽듯 읽어봐라”였지요.

처음 이 책을 읽을 때는 정말 괴로웠습니다. 거의 전혀 이해가 되지 않았기 때문이었죠. 책은 주로 설명하는 문장들로 이루어진 장(章)—최광무 교수님 말씀대로 진짜 수필 같은 장—과 수식(數式)들이 대부분인 장이 섞여 있는데, 구별 없이 이해가 곤란했습니다. 말로 설명하는 문장의 경우, 뻔히 알고 있다고 생각한 단어들로 이루어진 문장인데도 해석을 해 보면 사전적(辭典的)인 단어들의 잔해(殘骸)만 서로 유기적(有機的)인 관계가 없이 나열되어 서걱거리고 있을 뿐 내재(內在)한 의미는 거의 전달이 되지 않아서, 거듭하여 사전을 뒤적거리는 일이 되풀이되었습니다. 한 문장 내에서 뿐만이 아니라 문장과 문장, 단락과 단락 사이에서도 마찬가지로 상황이었습니다. 한편 수식들을 많이 사용한 장의 경우, 수식들의 기계적인

유도(誘導) 과정의 각 단계는 이해할 수 있었으나 그 직관적(直觀的)인 의미는 전혀 감(感)이 잡히지 않는 경우가 태반이었습니다. 당연히 억지로 몇 장을 읽은 뒤 포기하고 말았습니다.

이후 몇 달의 시간이 흐른 후 다시 도전을 해 봤습니다. 이번에는 훨씬 나왔습니다.<sup>5)</sup> 상당 부분을 이해할 수 있었습니다. 어설피게나마 이해하며 예닐곱 장(章)을 읽으면서 다익스트라가 말하고자 하는 바가 무엇인가 하는 점을 어렵듯이 깨닫게 되었고, 그 논지(論旨)는 (그 때까지 적어도 저와 같은 학부를 막 졸업한 학생들에게는) 보편적으로 들어보지 못한 것이었으나 한편으로 생각하면 너무나 맞는 것이었습니다. 상당한 충격이었습니다. 처음부터 다시 읽기 시작했고, 그러자 지난번까지 이해가 되지 않았던 여러 부분들이 자신의 의미를 드러냈습니다. 보는 관점에 따라 상대적으로 덜 중요하고 사소할지도 모르나, 기존의 여러 관념들을 깨뜨리는 언급들이 계속 나타났습니다. 개인적 취향에 따라 또 생각에 따라 조금씩 다르겠지만, 저는 다익스트라의 주장에 전폭적인 공감을 느꼈고 감복(感服)하였습니다.

저는 천재(天才)의 여러 특징들 중 하나가 대부분의 사람들이 당연하다고 생각해 온 기존 관념의 허구(虛構)를 깨뜨리는 정문일침(頂門一針)과 같은 한 마디의 말과 한 줄의 식이라고 생각합니다. 처음 들었을 때는 상식에 배치되는 것처럼 여겨지나, 다시 생각해 보면 옳다고 느끼게 되는 그런 말과 식입니다. 컴퓨터 과학 분야에 수많은 재능(才能)들이 있었고 있습니다만, 이러한 면에서는 다익스트라만큼 이 특징에 부합하는 경우도 없다고 개인적으로 생각합니다. 모두가 테스트를 통해 프로그램의 정확성(正確性, correctness)을 검증(檢證)하는 일을 당연하다고 생각하던 시절에 그는 테스트가 오류의 존재를 증명하는 데 유용하지 부재(不在)를 보이는 데는 거의 무용하다고 선언하였습니다. 모두가 당연히 분기문 없이는 프로그램을 작성할 수 없다고 생각하던 시절에 그는 분기문을 아예 프로그래밍 언어에서 삭제할 것을 주장하였습니다. 사소한(?) 예를 하나 더 들자면, 루프의 종료 조건을 느슨하게 하는 것이 바람직한 '방어적 프로그래밍(defensive programming)'의 한 예인 것처럼 많은 사람들이 생각할 때 그는 종료 조건을 가능한 한 강하게 해야 한다고 주장하였습니다. 모든 것이 그런 식입니다.

더욱이 다익스트라는 그 모든 주장을 최대한 신랄(辛辣)하게 합니다. (공격을 당하는 측은 괴로울지 모르나, 구경하는 사람들은 재미가 있지요.) 에두르며 말하지 않고 직격탄(直擊彈)을 날립니다. (틀림없이 다익스트라는 다른 사람들이 자신에게 그렇게 한다고 해서 불쾌해하지 않았을 것입니다. 논리적으로 맞는 주장이기만 하다면 말이죠. 그에게는 그러한 논쟁은 논리적이고 학술적인 것이지만 인간적인 것이 아니었겠지요.) 그래서 그의 글은 읽는 재미가 쏠쏠하고, 읽으면 읽을수록 우리나라 맛이 있습니다. 그가 평생 추구한 논리적/학문적 엄정성(嚴整性)과 더불어, 이러한 솔직 담백한 면이 그에게 인간적인 매력을 더합니다.

이러한 것들이 제가 다익스트라를 좋아하는 이유입니다. 다시 한번 더 고인의 명복을 빕니다.

5) 이 몇 달의 시간 동안 다익스트라의 책은 물론이고, 관련 있는 분야의 책이나 논문도 본 바가 없습니다. 이런 점을 생각하면 묘한데, '배후 작업(background job)'으로 머릿속 어딘가에서 이 책에 대한 처리가 이루어지고 있었다 봅니다. 푸앵카레(Jules Henri Poincare)가 말한 것이 맞는 것 같다는 느낌을 가끔 받습니다.