



# POPL/VMCAI 2013

2013년 1월 19일 - 2013년 1월 28일

서울대학교 허기홍

명실공히 프로그래밍 언어 분야의 이론 쪽 최고 학회인 POPL 에 다녀왔다. 매년 이 분야 최고 논문들과 교수들이 모이는 자리이다. 이번에는 특히 프로그램 위원장인 Radiha Cousot, 지도 교수이신 이광근 교수님을 비롯하여 프로그램 분석 분야,,의 태산 북두들이 주축이 된 해라서 더욱 흥미진진할 것으로 기대를 하며 참석을 하였다. 분석, 검증 분야 학회인 VMCAI 와 작년부터 시작한 PLMW (Programming Language Mentoring Workshop), 여타 워크샵도 함께 열려서 풍미를 더했다.



이번 POPL 은 이탈리아아 로마에 있는 한 고급 호텔에서 열렸다. 로마는 대학시절 여행을 간적이 한 번 있는데, 그 때 경험으로 말미암아 누구나 갖고 있는 이탈리아에 대한 선입견을 나도 갖게 되었다. 덥고, 지저분하고, 불친절하고, 물가는 비싼 이미지. 더구나 이번에 숙소를 예약하는 와중에, 이메일을 보내도 답장이 없고 전화를 하면 그제서야 메일함을 연 후 담당자가 없으니까 나중에 전화하라고 하니 기가막힐 노릇이었다. 허나 로마에 머무는 동안은 굉장히 즐거웠다. 수 천년간 그랬듯 도시는 여전히 우아했고, 식당이나 숙소나 길거리에서 만난 로마인들은 친절했다. 집집마다 특색있는 피자, 파스타, 젤라또 맛을 탐구하는 즐거움도 빼놓을 수 없다. 돌아오는 길에 곰곰이 생각해보니 이번엔 대학생 배낭여행객이 흔히 만나는 사람



과 환경을 마주하지 않은 이유가 큰 것 같다. 북적이는 기차역과 관광지, 그 주변의 공무원, 장사꾼, 무더위 따위.

## VMCAI 2013

POPL에 앞서 함께 열린 VMCAI 에 참가하였다. 다른 워크샵과 동시에 열려서 참가율과 집중도가 좀 떨어지는면이 있었지만 프로그램 분석 검증 분야에서 재미있는 논문과 발표들이 눈길을 끌었다.

### *Automatic Inference of Necessary Preconditions*

**Patrick Cousot, Radhia Cousot, Manuel Fähndrich and Francesco Logozzo**

함수의 필요 전조건 (necessary precondition) 을 자동으로 찾는 방법에 관한 논문이다. 이전까지 자동으로 전조건을 찾는 연구는 주로 충분 전조건 (sufficient precondition) 에 집중했다. 충분 전조건은, 그 조건 하에서 함수가 올바르게 실행된다는 것을 말해주는데 그 의미가 있다. 그러나 충분 전조건은 너무 보수적인것이 문제다. 그 함수를 부를 때 만족해야하는 조건이 대부분 너무 과하다. 반면 필요 전조건은, 그것이 만족되지 않았을때 함수가 항상 오동작 하는 조건을 말한다. 대개 충분 전조건보다 조건이 간단하다. 이 논문은 필요 전조건의 유용함을 역설하고 이를 자동으로 찾는 방법을 제시한다.

필요 전조건을 찾기 위해서는 우선 함수내에 있는 모든 제약 (assertion) 을 모으고 이 제약을 만족시킬수 있는 전조건을 후방분석으로 한단계씩 찾아나간다. 제약은 프로그래머가 직접 적어줄 수도 있고, 같은 저자들이 만든 정적 분석기인 cccheck 의 알람으로부터 만들어낼 수도 있다. 여기서는 세 가지 후방 분석을 보여준다. 함수의 시작부터 제약 지점까지 오는 모든 경로에

서 다 만족해야하는 조건을 위한 분석 (all-paths precondition analysis), 일부 경로에서 만족해야하는 조건을 위한 분석 (conditional-path precondition analysis), 정량자가 포함된 조건을 위한 분석 (quantified precondition analysis) 을 설명한다.

흥미로운 것은 이렇게 찾은 필요 전조건 중에서 동시에 충분 전조건인 것들이 제법 있다는 것이다. 즉, 몇몇 경우에는 이 분석으로 찾은 필요 전조건이 만족되면 그 함수가 올바르게 동작한다는 것이 확실히 보장된다는 뜻이다. 그러면 버그 검출기는 그 함수안에서 발생하는 경보를 모두 삭제하고 그 함수를 호출하는 곳에서 전조건을 만족하지만 보고함으로써 경보를 줄이는데 일조한다. 실험에 따르면 이러한 경우가 전조건을 자동으로 찾은 경우 중 27% 정도, 사람이 도와서 찾은 경우는 50%에 달한다고 한다.

굉장히 실용적인 동기에서 출발해서 제대로 된 분석기 (cccheck, clousot) 에 장착하고, 널리 쓰이는 소프트웨어 (.NET 라이브러리)를 대상으로 실험한 점을 본 받을 만한 논문이다. 반면에 여기서 이야기하는 핵심 내용 대부분이 같은 저자들의 VMCAI 2011 논문에 나와 있고, 이 논문은 그 논문의 보완인듯한 느낌이라 약간 아쉽다.

## *Hybrid Automata-based CEGAR for Rectangular Hybrid System*

**Pavithra Prabhakar, Parasara Sridhar Duggirala, Sayan Mitra and Mahesh Viswanathan**

하이브리드 오토마타를 좀 더 잘 모델링하여 CEGAR 기반 모델체킹을 잘하기 위한 연구이다. 하이브리드 오토마타는 연속적인 값과 이산적인 값을 함께 다루는 하이브리드 시스템을 모델링하는 대표적인 방식이다. CEGAR 는 모델 체킹 기법으로서 영성한 요약에서 시작하여 반례를 찾아내면서 점점 요약을 정교하게 해 가는 방식이다. 이전까지는 하이브리드 오토마타를 CEGAR 로 모델체킹할 때 변수 자체를 고려대상에서 제외하는 요약을 했다. 그리고 반례를 찾아내면서 필요한 변수를 고려하게끔 정교화 했다고. 허나 이 논문에서는 변수의 값을 구간으로 요약하여 CEGAR 의 성능을 더 높였다.

모델체킹 관련 발표를 듣다보니 제대로 공부해 보고 싶은 생각이 든다. 그 쪽에서 꾸준히 축적된 지식들을 우리 프로그램 분석에 유용하게 써먹을 수 있을지 모른다. 최근 여기저기서 듣기에, 분석에 들일 시간과 메모리는 얼마든지 있으니 더 정교한 것 없느냐?, 혹은 분석을 진행하면서 요약 도메인을 조절할 수 없느냐? 하는 질문을 받았다. 최근에 분석기의 정확도를 높이데 관심을 두고 있어서 더욱 자극적으로 들려왔다.

## PLMW

PLMW 는 PL 의 각 분야를 주도하는 연구자들이 그 분야를 30분동안 친절히 설명해주는 강연이다. 간혹 특정 연구 분야가 아니라 글쓰기, 연구 방법 등 일반적인 주제도 다룬다. 작년에 처음 시행되었다고 하는데 반응이 아주 좋았다고 한다. 올해 프로그램도 흥미진진하여 나를 포함한 많은 대학원생들이 좁은 강연장에 빼곡히 들어 앉았다. 때문에 같은 시간에 진행된 VMCAI 세션은 상대적으로 시들해지는 결과를 초래하였다. 주옥같은 강연 슬라이드는 홈페이지에 모두 매달려 있다. [1]

### *Programming languages and software verification*

**Xavier Leroy**

최근 정말 뜨겁게 달아오르는 분야인 프로그램 검증과 Coq 에 관한 소개였다. 지난 학기에 전산 논리와 Coq을 배우고, 연구실 동료들이 검증 관련연구를 하고 있던 터라 더 피부에 와닿았다. 프로그램 검증이 무엇을 하는 것인지, Coq 이 대강 어떻게 동작하는지를 보여주었다. 크게 기술적 내용은 없었다. 하지만 현재 POPL을 중심으로한 PL 학회에서 기계가 검증한 증명이 논문에 동봉되는 경우가 점점 늘어나고 있고 (POPL 2012 에는 이미 20%), 향후에는 일반화될 것이니 준비하라는 메시지는 모두에게 깊이 각인되었다.

### *What is a PhD? What is research? How do you succeed as a scientist?*

**Matt Might**

PL 전공자가 아니라도, 심지어 컴퓨터 전공자가 아닌 대학원생들도 가끔 그의 글과 자료를 읽어본 적 있다는 그 Matt Might 의 바로 그 내용이다. PhD 란 인류의 지식 확장에 조그마한 밑알이 되는 것, 그 이상 그 이하도 아니니 자만도 좌절도 하지 말고 정진하라는 내용. 우리가 연구하면서 내는 수많은 아이디어와 논문, 제안서는 대부분 쓰레기가 되겠지만 그런 진흙탕 속에서 진주를 발견할 수 있는 것이니 슬퍼말고 노력하라. 힐링 열풍이 대세인 요즘 시대를 반영하는 강연이다.

내용 중에서 가장 가슴에 와닿은 것은 아이디어를 기록하는 습관을 가지라는 부분이다. 연구를 하다가 문득 떠오르는 생각을 놓치지 말고 다 기록하라. 그 중 대부분은 결국 바보같은 생각으로 판명날지언정. 바보같은 아이디어를 많이 내본 사람이 훌륭한 아이디어를 낼 수 있다. 이것이 요지였다. 기록을 위해서 메모 프로그램인 EverNote 를 추천하기도 했다. 이 프로그램처럼 전화기, 컴퓨터, 태블릿 등에 자동으로 메모를 동기화 해주는 프로그램은 특히 공부하는 사람에게 필수인듯 하다.

Matt Might의 발표는 사람의 혼을 빼놓았다. 큰 화면에 핵심 한 두단어를 써놓고 청중을 자신에게 집중하게 한 후 화려한 언변으로 빨아들였다. 또 화려한 애니메이션과 기가막힌 삽화를 보



느라 시간가는줄 몰랐다. 30분 내내 웃음이 터지는 강연이었지만 그의 말이 너무 빨라서 많은 부분 같이 웃지 못한 것이 아쉽다. 듣다보니 든 생각이 있다. 이러한 발표 방식이 이 강연처럼 가벼운 내용을 다룰 때는 아주 적합한데, 진지하고 학술적인 내용을 다룰 때는 어떨지 모르겠다. 그의 학회 발표를 들은 적은 없지만 논문 발표 슬라이드를 보니 비슷한 형식이던데 궁금하다.

## *You and Your Research & The Element of Style* Phil Wadler

원래 공고된 제목은 “How to write a paper” 였는데 실제 강연은 위 제목이었다. 논문을 쓰는 것이 글을 쓰는 것 뿐 아니라 연구를 하는 것까지 포함하는 것이기 때문에 내용이 완전히 바뀐 것은 아니다. 재미있게도 “You and Your Research” 와 “The Element of Style” 은 이광근 교수님께서 연구실 신입생들에게 늘 언급하시는 강연과 책이다. 때문에 한 번 읽어본 내용이지만 다시 한 번 되새기며 들었다. 연사의 연극 하는듯한 몸짓과 대사도 강연의 맛이였다.

이 강연의 첫번째 부분은 해밍 코드로 유명한 리처드 해밍 (Richard W. Hamming) 박사의 “You and your research” 라는 강연의 중요 부분을 발췌한 것이다. 해밍 박사는 평생동안 일류 연구를 하는 훌륭한 과학자들을 관찰한 경험을 토대로 일류 연구는 어떻게 할 수 있는지를 이야기 하였다. 그 중 몇가지만 이 시간에 강조했는데, 핵심은 용기, 일반화, 소통로 요약할 수 있다. 1) 좌절하지말고 용기있게 밀어붙여라. 2) 답을 최대한 일반화하여 다른 누군가가 내 업적을 딛고 더 큰 업적을 이룰수 있게 하라. 그리고 나도 남의 업적을 딛고 올라서라. 3) 자기가 한 일을 기가막히게 설명하여 남의 관심을 끌어라. Wadler 교수님의 블로그 [2] 에 가면 해밍 박사의 실제 강연과 녹취록, 이광근 교수님의 홈페이지 [3] 에 가면 한글 번역문이 있다.

“Element of style” 은 정말 유명한 영어 쓰기 교본이다. 어렵지 않고 간단하지만 지켰을때 글이 깔끔해질 수 있는 규칙을 나열해 놓았다. 읽어보면 쉽게 수공이 가는데 막상 글을 쓸 때는 잘 지키지 못하는 경우가 많다. 머리로만 이해하고 깊이 새기지 않아서인데 이 참에 다시 한 번 읽어보아야 할듯 싶다.

## 로마와 이탈리아 음식

여름 로마는 너무나 더워서 사람을 지치게 만들었는데 겨울 로마는 쾌적하고 싱그러웠다. 매 끼 이탈리아 음식을 맛볼 수 있다는 것도 행복이었다. 학회장 바로 옆에는 “보르게세”라는 아주 큰 공원이 있었다. 얼마나 오래되었는지 모르겠지만 나무의 높이가 예사롭지 않았다. 학회 쉬는 시간에 답답한 숨통을 틔워보려 가끔 들렀더니 달리는 개와 산책하는 사람, 울창한 나무가 기막히게 어우러져있었다.





날씨가 좋아서 로마 시내를 늘 걸어서 누볐다. 관광지 주변을 지날때면 심심찮게 거리의 행위 예술가들을 만났다. 대부분 식상한 모습이었지만 하나 눈길을 끈 것이 있었다. 승려복장을 한 두 사람이었는데 한사람이 앉아서 막대기를 들고 있고 한 사람은 그 막대기 위에 올라 앉아있다. 정말 평온한 모습으로 도담듯 공중부양해 있는 모습이 한편으로는 익살스러웠다. 그날 본 예술 중 제일 재미있어서 유일하게 동전을 몇 닢 넣어준 팀이다.

이탈리아 사람들은 맨날 피자과 파스타만 먹는지 길거리 간판에는 온통 피자뿐이었다. 우리 나라에 온 외국인들도 그렇게 생각할까? “한국인은 맨날 밥만 먹는지 어느 식당에를 가도 항상 쌀밥이다.” 허나 같은 피자라도 집집마다 맛이 다르고, 그 집 이름을 건 피자나 파스타가 있어서 지루하지 않았다. 학회장은 관광지에서 좀 떨어진 곳이라 본연의 맛, 색다른 맛을 즐길 수 있었지만 관광지로 갈수록 맛은 평범해졌다. 학회 첫날이 일요일이어서 학회장 주변 음식점이 대부분 문을 닫았더라. 닫힌 문 앞에서 지도를 펼쳐놓고 머리를 맞대로 있으니 근처 주유소 아저씨가 문 연 집을 하나 소개해 주었다. 그 아저씨는 그 이후로도 가끔 우리의 네비게이션 역할을 해준 고마운 분이다. “Casa Mia” 라는 곳인데 아저씨가 아주 유쾌하고 친절했다. 돌아오는날까지 우리는 그 집에 세 번이나 가게 된다. 나중에는 아저씨가 와인도 공짜로 주는 친절도 베풀어 주었다.

젤라또와 초콜렛은 맛난 간식거리다. 예전에 로마에 왔을때 맛있게 먹었던 백년 전통 젤라또집이 생각나 숙소에서 40분을 걸어서 두 번이나 갔는데 두 번 다 문이 닫혀있었다. 아쉽지만 다른 젤라또 집에서 먹은 것으로 만족해야했다. 포항공대를 졸업한 임현승 형을 학회장에서 오랜만에 만났더니 “Pocket coffee” 라는 것을 소개해 주었다. 한국인들이 로마에 오면 기를 쓰고 사 담아가는 초콜렛이라고 하는데 초콜렛안에 에스프레소가 담겨 있는 것이 특징이다. 페르미니 역 근처 편의점에서 판다는 정보를 알려주고는 같이 사러가지는 못했다. 우리는 학회 후에 페르미니 역 근처 편의점에 물사러 갔다가 우연히 발견하여 그 집에 있는걸 다 쓸어왔다. 현승형은 잘 사서 돌아가셨는지 궁금하다.



## POPL 2013

드디어 PL 분야 최고학회라는 POPL 이 시작되었다. POPL 에서 두드러진 특징은 아무래도 Coq 이었다. Coq 증명을 편하게 할 수 있는 연구는 물론이고 초청강연과 다른 연구 발표에서도 심심치않게 Coq에 관한 이야기가 나왔다. 많은 이들이 자기가 만든 시스템이 올바르다는 것을 Coq으로 증명하고 홈페이지에 매달아 놓았더라. 바람직한 방향인 것 같고 내 연구에도 필요가 있으면 언젠가 시도해야 하겠다.

### *Inductive data flow graphs.*

**A. Farzan, Z. Kincaid, A. Podelski**

동시성 프로그램 (concurrent program) 을 호아 논리 (Hoare logic) 으로 검증할 때 효율성을 높이기 위해 iDFG (inductive data flow graph) 를 도입한다. 동시성 프로그램을 검증할때는 여러 스레드가 수행되는 순서를 모두 고려해야 하기 때문에 일이 매우 커진다. 하지만 프로그램의 명령어 자체가 아닌 명령어의 의미를 두고 보면 고려해야하는 상황이 훨씬 줄어들 수 있다. 즉 핵심은 구문의 의존성을 보는 것이 아니라 의미의 의존성에 집중하는 것이다. iDFG는 이러한 의미 의존성을 바탕으로 그래프를 그리고 각 간선에는 호아 논리 증명식을 덧붙여놓은 것이다.

검증 알고리즘은 이렇게 동작한다. 1) 동시성 프로그램에서 한 실행 경로를 선택한다. 2) 정적 분석기를 이용해서 그 실행 경로의 올바름을 검증하는 호아 논리 증명을 뽑아내고, iDFG 를 만든다. 3) 지금까지 축적한 iDFG 와 새 iDFG를 합친다. 4) 축적한 iDFG가 모든 프로그램 실행을 포섭하는지 검사한다. 그렇다면 프로그램은 올바른 것이고 아니면 1 부터 다시 반복한다. 이렇게 함으로써 동시성 프로그램 모델 체킹에서 특히 상태 폭발이 심한 현상을 4 번에서 포섭을 검사하는데 들이는 노력으로 바꾸어낼 수 있다.

프로그램의 구문이 아닌 의미에 집중하여 의존성을 만들고 효율을 높이는 데서 우리 PLDI 2012 에 발표한 스파스 분석 (sparse analysis) 과 비슷하다. 실제로 이 논문에서도 DFG를 순차적 (sequential) 프로그램 분석에 사용한 관련 연구로 우리 논문을 간단히 언급했다. 이런 것을 보니 여러 사람들이 비슷한 일을 다른데서 다른 방식으로 하고 있는 것 같다. 내가 하는 분야를 우선 깊숙이 파고 단단히 선 후에 주위에 귀를 기울이면 더 흥미로운 일이 분명 벌어질 것 같다.

### *Fully abstract compilation to JavaScript*

**C. Fournet, N. Swamy, J. Chen, P. Dagand, P. Strub, B. Livshits**

자바 스크립트를 날로 쓰면 엉망진창이 된다는 것은 이제 널리 알려져있고, 어떻게 해서든 안전하고 편하게 쓰려는 노력이 많은데 그 중 꽤 재미있는 연구이다. ML과 비슷한 f\* 라는 함수형

언어로 프로그램을 작성하고 이를 자바 스크립트로 컴파일해서 실행하는 것이다. 이러한 시도가 처음은 아니라고 한다. 하지만 그 전과 다르게 이 연구에서는 컴파일 과정이 올바르고 결과물이 보안에 취약하지 않다는 것을 보장한다.

시스템은 이렇게 구성되어있다. 컴파일러는 우선  $f^*$  로 작성한 프로그램을 의미가 동일한 자바 스크립트 프로그램으로 변환한다. 이렇게 만들어진 자바스크립트의 보안 취약점을 제거하기 위해서 인터페이스를 포장한다. 자바스크립트의 동적 타입을 이용해서 외부 라이브러리가 여러가지 악성 공격을 할 수가 있다. 이를 막기 위해 인터페이스에서는 타입을 강제해주는 일을 제공한다. 이렇게 만들어진 자바스크립트 프로그램이 올바름을 보이기 위해서는 추가적인 검산이 필요하다. 그 때문에 포장된 자바스크립트 프로그램을 다시  $f^*$  프로그램으로 바꾸고 원래 만든  $f^*$  프로그램과 의미상 동일하다는 것을 검산한다.

뚜렷한 문제의식, 간단 명료한 아이디어, 탄탄한 이론과 구현이 어우러진 좋은 연구라고 생각한다. 논문을 읽어보니 자바스크립트의 악성 공격을 구체적인 예로 들면서 자기들의 방법을 설명해가고 있다. 쉬운 예로 설명해주니 더욱 이해가 잘 가서 재미있게 읽은 논문이다.

### ***Optimizing data structures in high-level programs: new directions for extensible compilers based on staging.***

**T. Rompf, A. Sujeeth, N. Amin, K. Brown, V. Jovanovic, H. Lee, M. Jonalagedda, K. Olukotun, M. Odersky**

이 논문은 단계 변환 (staging) 을 이용하여 복잡한 자료구조를 다루는 고수준 언어 (high-level language) 프로그램을 최적화하는 방법을 제시한다. 보통 이런 자료구조는 요약 수준이 높아서 최적화하기가 쉽지 않은 것이 문제이다. 이런 문제는 보통 두 가지 방식으로 접근한다. 하나는 매크로 시스템이나 부분 실행 (partial evaluation) 을 사용하여 실제 컴파일시점 이전에 가능한 모든 요약을 풀어헤친다. 이는 단순히 구문 조작으로 가능하지만 할 수 있는 일이 적다. 또는 복잡한 자료구조를 위한 배경지식을 컴파일러에게 미리 주입해놓을 수도 있다. 이 방식은 컴파일 중간에 변환 단계를 하나 더 삽입함으로써 가능한데 이것이 쉬운일이 아니다. 때로는 독립적인 변환 단계들이 서로 간섭을 일으켜 변환이 올바르게 되지 않기도 한다. 여기서는 단계 변환을 이용하여 컴파일러 최적화 경로를 정의한다. 그러면 기존에 IR 변환으로 구성하던 것에 비해 구현이 편리하고 변환 단계끼리 간섭을 염려하지 않아도 된다.

함수형 언어의 성능을 고민하는 문제의식과 다단계 언어를 컴파일러 최적화 단계에 이용한 아이디어가 재미있는 논문이었다. 우리 연구실에서 다단계 언어 관련하여 주옥같은 논문이 나왔는데 별로 관심을 갖고 있지 않던 터라 더욱 자극적이었다. 허나 본문을 읽어보니 아홉명이나 되는 저자들이 쏟아놓은 수많은 소단락들을 따라가기가 버거웠다. 논문을 쓸 때 어느 정도 내용을 넣을 것인가를 항상 고민하게끔 한다.





## 암스테르담의 명주, 하이네켄

돌아올때는 로마를 떠나 암스테르담에서 인천행 비행기를 갈아타게 되었다. 갈아타기까지 다섯시간 남짓되는 애매한 시간이 있었는데, 네덜란드는 처음인데다 특히 스키폴 공항이 도심에서 가깝다는 소식에 학주형과 무작정 공항을 나와 도심행 기차표를 끊었다. 튤립과 치즈로 유명한 나라라더니 듣던대로 공항 면세점이든 기차역이든 온통 튤립, 치즈를 팔고 있더라. 치즈는 그렇다쳐도 튤립을 면세점에서 사가는 사람이 있을지 궁금하다.

역에서 얻은 여행 지도를 보니 근처에 하이네켄 양조장과 박물관이 있다고 하여 발걸음을 거기로 재촉했다. 도시 구석구석을 연결해 놓은 운하를 넘고넘어, 하이네켄이라고 쓰인 커다란 광고판을 보고 다운줄 알고 사진 찍고, 길을 물어물어 양조장에 도착했을때는 비행기 출발 시간 약 세 시간 전이었다. 박물관에는 견학 프로그램이 있고, 프로그램에 참여하면 하이네켄을 한 잔 준다고 들었다. 헌데 한 시간 반짜리라 참가하지 못했고, 아쉬우나마 맥주만 한 잔 먹고 갈 수 없냐고 물었더니 박물관 옆에 있는 작은 술집이 훌륭하다며 추천해 주었다. 좁은 술집 바에 걸터앉아 마신 하이네켄은 정말 최고. 근본없는 몇몇 국산 맥주에는 비교하기조차 부끄럽고, 동네 술집에서 먹던 하이네켄 생맥주의 알 수없는 역겨움 대신 부드러운 고소함이 거기 있었다. 비결은 보리의 양과 질, 생맥주 기계의 위생상태, 그리고 기분탓일 것이다.

## 맺음말

학회에서 새로운 것을 보고 듣고, 알던 것도 자극 받고는 공부해야할 것을 한보따리 싸들고 왔다. 새 얼굴, 반가운 얼굴을 만난것도 큰 기쁨이었다. 이 글을 빌어 늘 학생들에게 늘 좋은 기회를 주시는 이광근 교수님께 고맙다는 말씀을 드리고 싶다. 또한 학회 출장시 여러 가지를 도와주시는 소프트웨어 무결점 연구센터와 센터 행정실분들께도 고마운 마음을 가지면서 여행기를 마무리 한다.



## 참고자료

[1] <http://www.doc.ic.ac.uk/~gds/PLMW>

[2] <http://wadler.blogspot.co.uk>

[3] <http://ropas.snu.ac.kr/~kwang>