컴퓨터과학이 여는 세계, 과제 1

Exercise1 “400 년의 축적”

3 세기 동안의 티끌이 모아 만들어진 태산, 컴퓨터

교과명 : 컴퓨터과학이 여는 세계

지도교수 : 이광근 교수님

학과 : 영어영문학과

이름 : 정선이

학번 : 2012-

제출일 : 2013.04.08

1. 서론

지난 달 20 일에 사이버 테러로 일부 금융기관, 방송국, 통신사의 전산망이 마비되었다. 전산망이 마비되는 동안 금융거래가 중단되고 방송을 할 수 없는 전국적인 혼란이 왔다. 이 3.20 사태는 컴퓨터가 우리의 삶 속에 얼마나 깊게 관여하고 큰 영향을 미치는지 단적으로 보여주는 사건이었다. 이제는 컴퓨터 없이는 편안하게 살 수 없고 오히려 컴퓨터에 의해 우리의 삶이 조종되기도 하는 시대에서 컴퓨터의 기원에 대해 알아보고 이를 바탕으로 컴퓨터를 이해하는 것은 의미 있는 일이다.

인류의 위대한 발명품으로 불려지는 컴퓨터는 20 세기 수학계의 좌절을 재확인하는 것에 쓰인 소품을 모태로 하고 있다. 수학자들의 좌절은 1931 년, 미국의 수학자 쿠르트 괴델이 "기계적인 방식만으론 사실인지 판정할 수 없는, 그런 명제가 존재한다." 라는 것을 증명함으로써 시작되었다. 이 같은 괴델의 증명이 있기 전까지, 라이프니치를 시작으로 기계적인 방식으로 모든 수학적인 사실들을 만들어 낼 수 있을 것이라는 희망을 품고 있었다. 그런데 괴델의 증명으로 수학자들은 희망대신 좌절을 안게 된 것이다. 튜링은 이 절망을 새롭게 증명하는 과정에서 테이프, 헤더기, 작동 규칙표로 이루어진 간단한 기계부품들을 정의했다. 그 후, 이 부품들로 하나의 특별한 기계를 만들어 보이는데, 그 기계가 바로 컴퓨터의 조상인 것이다.

그런데 컴퓨터는 앨런 튜링에 의해서만이 아니라 그 이전의 수많은 수학자들과 공학자들의 노력이 쌓여 탄생한 것이다. 튜링의 "보편 기계"가 창안되는 것에 직접적인 영향을 끼친 배경은 힐베르트의 결정 문제와 괴델의 불완전성 정리가 있다. 그리고 이들의 생각과 문제를 이해하기 위해서는 현대 논리학을 시작했다고 볼 수 있는 프레게와 집합론의 창시자인 칸토어를 이해해야 한다. 게다가 현대 컴퓨터와 현대 논리학의 가장 기본적인 생각을 제시한 라이프니츠와 라이프니츠의 생각을 조금은 독립적으로 발전시켰던 조지 불을 살펴보아야 한다. 본 리포트에서는 현대 컴퓨터의 시작을 알리는 보편 기호 체계를 제시한 라이프니츠를 시작으로 컴퓨터의 기원에 대해 이야기를 풀어나갈 것이다.

2. 본론

2.1 컴퓨터에 대한 생각의 흐름을 연 라이프니츠

라이프니츠가 "보편 기호 체계"를 제시하는 것에 큰 영향을 준 것은 아리스토텔레스의 논리학 체계이었다. 아리스토텔레스는 개념들을 고정된 범주로 분할하였다. 이런 생각은 라이프니츠가 소리가 아니라 개념을 표현하는 원소들로 이루어진 특별한 문자 체계를 찾도록 하였다. 이런 문자 체계에 바탕을 둔 언어는 그 언어로 쓰여진 문장들이 참인지를, 그리고 문장들 사이에 존재하는 논리적 관계가 무엇인지를 기호 연산을 통해 결정할 수 있는 것이다.

라이프니치는 아무런 뜻이 담겨 있지 않은 한글 자음, 모음이나 알파벳 기호들과 달리 자연스럽고 적절한 방식으로 어떤 분명한 생각을 담고 있는 기호의 체계를 실제 기호 체계라고 명명하였다. 미적분은 모든 기본 개념을 표현하는 문자 체계를 세우고자 했던 라이프니츠의 생각의 모델이다. 라이프니츠는 미적분 기호∫와 d 는 한글이나 알파벳처럼 의미 없는 소리를

나타내는 것이 아니라 개념을 나타낸 것이다. 그리고 그는 한층 더 나아가 실제 세계의 기호 체계일 뿐만 아니라 인간의 모든 사고 범위를 포괄하는 “보편 기호 체계”를 생각하였다.

라이프니츠는 인간이 생각하는 것을 정학하게 나타내는 알파벳과 이런 기호를 다루는 데 알맞은 연산 도구를 설계하기 위해서는 세 가지 주요한 구성 요소가 필요하다고 보았다. 첫째로, 적당한 기호들을 고르기 전에 인류가 지닌 모든 범위의 지식을 포함하는 백과사전을 만드는 것이 필요하다. 다음으로 기본이 되는 핵심 개념들을 선택하고 그것들 각각에 알맞은 기호를 부여하는 일이 가능함을 증명해야 한다. 마지막으로 그것을 증명하면 연역법이 이 기호들의 조작으로 환원할 수 있다. 이것이 “추론 계산법”으로 오늘날의 기호 논리학이다. 이런 추론 계산법의 실례로 일반 대수에서 수를 다루는 규칙을 서술하는 방식과 같은 방식으로 논리적 개념을 다루는 규칙을 서술하게 될 논리 대수를 제시하였다.

2.2 컴퓨터 전기 회로의 기반이 되는 불 대수, 조지 불

조지 불은 의도치 않게 라이프니츠가 찾고자 했던 추론 계산법을 마련해 준다고 말할 수 있는 “불 대수”를 만들었다. 불 대수는 논리학을 대수적인 방법으로 해명하려고 시도한 것으로 불 대수에서는 하나의 낱말에 의해 서술되는 개체들 모두를 하나의 "집합"으로 보았다. 그리고 불은 숫자를 나타나는데 문자를 사용해 온 것과 마찬가지로 집합을 표현하는 데 문자를 사용하였다. 예를 들어 x 는 오직 흰 것들의 집합이고 y 는 말을 대표하는 각각 특정한 집합을 나타내는 것이다. 예를 들어 xy 는 흰 말들의 집합으로 교집합에 대한 생각으로 까지 나아가게 된다. 그는 대수에서의 곱셈을 교집합으로, 덧셈을 합집합으로, 그리고 뺄셈을 차집합으로 상정하고 논리적 추론을 집합적 대수 표현으로 환원시켰다.

그러나 일반 대수와 다른 것은 일반 대수에서는 $4\times4=16$ 이고, $4+4=8$ 인 반면에, 불 대수에서는 $xx=x$ 이고 $x+x=x$ 라는 점에 있다. 이것의 의미는, x 의 의미를 위에서 언급한 것과 같다고 보면 xx 의 의미는 단지 흰 것들의 집합일 뿐인 것이다. $xx=x$ 를 만족하는 수는 0 과 1 뿐이나. 그는 0 과 1 이라는 기호를 집합적으로 재해석하게 되었고, 그는 0 은 공집합을, 1 은 전체 집합을 표현한다고 해석하기에 이르렀다. 더 나아가 0 은 명제가 거짓임을, 1 은 명제가 참임을 의미하도록 하여 명제의 참 거짓도 대수적으로 표현하였다.

그런데 논리학은 완전성을 목표로 하기 때문에 모든 수학적 추론을 포함했는지 여부가 중요하다. 그러나 불의 대수를 이용한 추론은 완전성을 이루지 못한다. 예를 들어 “모든 성공한 학생은 똑똑하거나 부지런하다.” 라는 문장에서 성공한 학생 집단을 X 로 똑똑하거나 부지러한 학생들의 집합을 Y 로 한 개의 단위로 표현하면 “모든 X 는 Y 이다.” 이다. 그런데 이 추론은 똑똑함 때문에 성공한 학생들과 부지런함 때문에 성공한 학생들을 구별해 내는 어떤 추론도 허락하지 못하는 문제가 생긴다.

2.3 컴퓨터 프로그램에 쓰이는 개념 표기법, 프레게

고틀로프 프레게는《개념 표기법》에서 불 대수의 한계를 극복하며 더 세밀한 종류의 추론을 다룰 수 있는 논리 체계를 세운다. 프레게는 논리적 관계를 나타내는 자신이 만든 특수한

기호들을 도입하였다. 기호에는 ¬(아니다),∨(합집합),∧(교집합), ∀(모든),⊃(만일…이면…이다), ∃(어떤)가 있다. 그의 이런 기호는 라이프니츠의 보편 언어 개념을 지침으로 삼아 만든 것으로 라이프니츠의 영향력이 계속되고 있음을 알 수 있다.

프레게는 모든 수학 분야가 논리학에 토대를 둘 수 있음을 보이려 하였다. 그 과정에서 논리를 사용하지 않고 자신의 논리를 전개하는 방식을 찾아야 했다. 그가 채택했던 방법은 개념 표기법을 구문론이라는 엄격하게 정밀한 규칙을 가진 인공 언어로 발전시키는 것이었다. 이를 통해 논리적 추론을 기호들이 배열된 방식에만 관련 있는 순수하게 추론 규칙으로 나타내는 것이 가능하게 되었다. 이런 관점에서 개념 표기법은 오늘날 사용되는 모든 컴퓨터 프로그래밍 언어의 조상이다. 그리고 수리 논리학의 정확한 체계가 수학자들이 일반적으로 사용하는 모든 추론을 포괄하게 된 프레게의 논리학은 엄청난 내용의 연구를 위한 기초가 되었고 간접적으로는 앨런 튜링이 만능 컴퓨터에 대한 착상을 가지도록 해 주었다.

그러나 프레게의 논리학에는 한계점이 있었다. 버트란트 러셀은 뒤에 칸토어의 무한 기수에 대해 생각하다가 이러한 오류를 발견해냈다. 러셀에 따르면 한 집합이 자기 자신의 원소이면 예외적이라고 하고 그렇지 않다면 일반적이라고 할 때, 일반적 집합들의 집합 E 는 어느 쪽에도 속하지 못한다. 어느 쪽을 택해도 논리는 모순에 빠지게 되고, 이에 따라 프레게 논리 체계가 전제로 삼았던 모순율이 성립하지 않는다는 것이 밝혀졌다. 그리고 라이프니츠 이후 수학계가 라이프니츠의 보편 언어를 구축하기 위한 흐름이었다는 것을 감안하면, 라이프니츠는 논리적 연역을 할 수 있을 뿐만 아니라 과학과 철학의 모든 진리도 자동적으로 포괄하는 언어를 원하였지만 프레게의 개념 표기법은 논리적 연역만 할 수 있는 한계가 있다. 게다가 프레게의 규칙은 원하는 어떤 결론이 주어진 전제들에서 연역될 수 있는지 결정할 아무런 계산 절차도 제공하지 못한다.

프레게의 논리학에는 이러한 한계점들이 있지만 앨런 튜링이 가능한 어떤 계산도 그 자체로 수행할 수 있는 하나의 단일한 보편 기계를 고안하는 것이 원칙적으로 가능하다는 것을 알아내는 것에 큰 도움을 주게 된다. 그의 이런 생각은 프레게의 논리학에서 제안된 추론이 올바른지를 보여 줄 수 있는 방법이 존재하지 않는다는 것을 증명하는 과정에서 얻게 된 것이다.

2.4 **무한 집합론과 대각선 방법을 제시한, 칸토어**

1936 년 등장할 앨런 튜링의 논문에서의 증명이 어떻게 이루어졌는지를 이해하기 위해서는 칸토어와 그의 대각선 정리를 반드시 짚고 넘어가야 한다. 대각선 정리는 괴델의 불완전성 정리와 튜링의 멈춤 문제 해결 불가능성 정리에도 사용되었기 때문이다. 칸토어는 그 당시 신의 영역이라고 생각되는 무한 특히 실무한에 대한 수학 이론을 창조하였다. 삼각 함수를 연구하던 칸토어는 생각을 더 넓혀서 무한 집합을 완결된 전체로 다루고 그것에 관해 복잡한 계산을 수행하려고 하였던 것이다. 그리고 칸토어는 이러한 무한에는 다양한 크기가 존재한다는 사실을 우리에게 깨우쳐주고 있다. 여기서부터 칸토어의 집합론이 시작되는 것이다.

N_0 N_0 그는 두 집합이 같은 수의 원소를 가졌다고 결론을 내리기 위해서는 일대일 대응이 두 무한 집한 사이에서 정의될 수 있어야 한다고 생각했다. 칸토어는 서로 다른 두 무한 집합 사이의 일대일 대응이 가능하다는 데서 연구를 시작하였다. 그는 일대일 방식으로 분수들의 집합과 자연수의 집합 사이의 일대일 대응, 대수적 수와 자연수 간의 일대일 대응을 밝혔다. 그러나 자연수의 집합과 모든 실수의 집합의 일대일 대응이 가능하지 않다는 것을 밝히며 실수 집합의 기수 C 는 자연수 집합의 기수 보다 크다고 그는 주장했다. 그리고 그는 C 와 사이에는 어떠한 다른 기수도 없다고 주장했는데, 바로 이것이 칸토어의 연속체 가설이다.

	0	1	2	3	…
S_0	**+**	−	+	−	…
S_1	+	**+**	+	−	…
S_2	−	+	**+**	−	…
…	…	…	…	…	…

N_0C>라는 사실은 대각선 방법을 통해서 증명될 수 있는데, 다음의 표를 보며 설명할 수 있다. 표에서 S_0, S_1 , S_2 …로 표현되는 S_n 은 자연수의 부분집합을 표현하는 것이다.

위의 표에서는 자연수의 부분집합과 자연수가 각각 일대응 대응을 이루고 있는 것을 볼 수 있다. 그런데 우리는 위의 표에서 굵은 색으로 표현된 것들로 이루어진 새로운 집합 S_V 을 만들 수 있다. 문제는 S_V N_0 N_0 위의 표에 나타나지 않는다는 것이다. 이를 통해 이러한 모든 자연수 집합의 기수는 보다 클 것이라는 사실을 알 수 있다. 이때 모든 자연수 집합들의 집한 기수가 실수 집합의 기수 C 임을 증명하는 것은 가능하므로 우리는 C>라는 사실을 대각선 방법을 통해 유도할 수 있다. 칸토어의 대각선 방법은 후에 튜링이 튜링기계가 자동으로 풀 수 없는 멈춤 문제를 증명하기 위하여 쓰이게 된다.

그러나 칸토어가 여기에서 더 나아가 초한 기수 또는 초한 서수를 모두 모아 하나의 집합으로 만들려는 시도로 인해 수학의 기초가 위기에 빠지게 된다. 만약 모든 기수의 집합이 있다면 그 집합의 기수는 무엇인가에 대한 문제에 부딪히게 된다. '모든 집합들의 집합이 있을 수 있는가?'하는 의문을 가진 버트란트 러셀이 자기 자신의 원소가 아닌 모든 집합들의 집합이라는 역설을 찾아냄으로써 수학의 본체가 흔들린 것이다. 이런 칸토어의 무한을 둘러싸고 벌여진 논란 속에서 만능 컴퓨터에 대한 앨린 튜링의 수학적 모델이 나오게 된다.

2.5 산술의 무모순성을 증명하고자 한, 힐베르트

러셀의 역설로 인해 20 세기 초에는 '수학의 위기'가 찾아온다. 이런 수학의 위기를 정면 돌파하려고 했던 인물이 바로 힐베르트이다. 힐베르트는 위태한 수학 기초론을 다시 세우려 하였다. 그는 기하학에 관한 공리들의 집합을 제시했는데 이는 유클리드의 고전적 방법의 허점 몇 가지를 메워 주는 것이었다. 마침내 힐베르트는 자신의 공리들에서 어떤 모순도 유도될 수 없다는 증명을 하였다. 이 증명은 기하학에 관한 그의 공리 체계에 어떤 모순이 있게 되면 실수의 산술에서도 모순이 생기게 된다는 것을 보여 주었다. 그래서 그가 한 일은 산술의

무모순성 문제를 남겨 둔 채, 유클리드 기하학의 무모순성을 산술의 무모순성으로 환원한 것이었다.

그러나 힐베르트는 이제 실수 산술 체계의 무모순성을 밝혀야 할 때가 왔다고 생각했다. 이를 위해서는 그는 어떠한 직접적 방법론이 필요하다는 것을 깨우쳤는데, 바로 그것이 그가 주장한 '메타수학 프로그램'이다. 그리고 힐베르트가 설계한 프로그램에는 내부와 외부 두 부분으로 나누어 볼 수 있다. 프로그램을 내부에서 보면 그것은 각각의 작은 연역적 단계가 엄밀하게 만들어져 있는 수학이다. 그러나 외부에서 보면 의미를 고려하지 않은 여러 가지 식과 기호 조작일 뿐이다. 힐베르트의 프로그램은 형식 체계 내부에서는 모든 종류의 수학적 방법들이 완전히 제한 없이 사용되도록 허용되지만, 메타수학적 방법들은 힐베르트가 유한의 방법이라고 일컬은 방법들로 제한되어야 했다. 그리고 그는 그의 방법에 칸토어의 초한수도 포함시켰다.

힐베르트의 프로그램은 산술의 무모순성을 가장 주요한 문제로 삼았다. 이 문제에 관해 연구하여 얻은 결과를 담은 그가 쓴 교과서에는 프레게의《개념 표기법》의 기초 논리학에 대한 두 가지 문제를 담고 있다. 첫 번째 문제는 체계 외부에서 보았을 때 어떤 타당한 논리식이 교과서에서 제시된 규칙들만 사용해서 체계 내부에서 도출될 수 있음을 증명하는 것이다. 두 번째 문제는 "힐베르트의 결정 문제"이다. 기초 논리학의 어떤 논리식이 주었을 때, 유한한 단계들을 거치면서 그 논리식이 타당한지 아닌지 결정할 방법을 제공하는 문제이다.

2.6 **결정 불가능한 명제를 증명한, 괴델**

괴델은 힐베르트가 남긴 문제인 PA 의 무모순성을 증명하는 과정에서 이것을 증명할 수 없다는 것을 증명하게 된다. 이것에 대해 생각하다가 형식 논리 체계를 내부에서 본 힐베르트와 달리 외부에서 바라보는 것의 중요성을 깨닫게 된다. 그는 왜 메타수학이 어떠한 형식 논리 체계 내부에서 직접 전개될 수 없는 지에 대해 의문을 품었다. 내부 명제들의 부호화를 통해 외부를 내부로 가져올 수 있다고 괴델은 생각했다. 그리고 이러한 과정에서 궁극적으로 그는 외부적 관점에서 바라보면 참이라고 할 수 있지만, 체계 내부에서는 결코 증명할 수 없는 명제가 존재한다는 사실을 깨우쳤다.

괴델은 칸토어의 대각선 방법을 이용하여 증명 불가능하다고 주장되어지는 명제와 바로 그 주장을 하는 명제를 일치시킴으로써 그러한 명제를 우리에게 제시한다. U 라는 명제가 "U 는 PM 안에서 증명될 수 없다."라는 명제라면 이 명제는 참이지만 체제 내부에서 결정 불가능한 특징을 지니게 된다. 여기서 U 는 U 도 U 의 부정도 PM 안에서 증명가능하지 않다는 속성을 지니는 것이다. 이 명제는 외부 관점에서 보면 참이라는 사실이 분명하다. 이러한 문제를 '결정 불가능한 명제'라고 부른다. 이처럼 괴델은 어떤 체계의 무모순성은 그 체계 내부로부터 증명 불가능 하다는 사실을 증명해냈으며, 이로서 무모순의 규칙들을 찾아내기를 원했던 힐베르트 프로그램은 실패하게 되었다.

그는 또한 앞서서 제기된 칸토어의 연속체 가설 역시 결정 불가능하다는 사실을 밝혀냈다. 괴델의 이러한 주장은 기계적인 방법으로 모든 명제들의 참·거짓을 판명할 수 있지

않을까하는, 라이프니츠 이후로 줄곧 내려온 수학계의 큰 꿈을 좌절시키는 것이었다. 그는 이로서 “기계적인 방식만으로는 사실 판정이 불가능한 명제가 존재한다”는 사실을 밝혀낸 것이었다.

2.7 400 년 업적들의 결정체 튜링기계, 튜링

당시 24 세였던 앨런 튜링은 괴델이 증명했던 방법 이외의 다른 방법으로 어떻게 힐베르트가 꿈꿨던 일반적 알고리듬은 존재하지 않는다는 사실을 증명하였다. 이러한 색다른 증명 과정상의 부산물로 앨런 튜링은 현대 컴퓨터의 조상을 이루어낸다. 앨런 튜링은 “기계적”이라는 것이 무엇인지를 정의하기 위해 튜링 기계를 상정했던 것인데, 그는 모든 기계적 계산이 무엇인지를 정확히 정의하는 과정에서 보편 만능의 기계(튜링 기계)를 생각해 냈던 것이다. 이 보편 만능의 기계는 임의의 튜링 기계의 규칙표를 입력으로 받아, 정확히 동일한 계산을 수행할 수 있는 단 하나의 튜링 기계를 의미하는 것이었다. 이 튜링 기계는 이로서 튜링이 정의한 ”모든 기계적 계산“을 할 수 있는 것이었다.

그리고 튜링 기계는 괴델에 이어 힐베르트가 꿈꾸던 결정 문제를 해결할 수 없음을 보여주고 있다. 튜링 기계는 입력값을 받아 어떤 수들에 대해서는 멈출 것이고, 어떤 수들에 대해서는 멈추지 않을 것이다. 이때 우리는 전자의 경우에 해당하는 자연수 집합을 그 특정 튜링 기계의 멈춤 집합으로 볼 수 있고, 한 튜링 기계의 멈춤 집합이 한 개의 집단을 이룬다고 생각한 뒤 그 기계의 부호수를 그 기계의 이름표로 생각할 수 있다. 이때 우리는 대각선 방법을 이용함으로써 어떠한 튜링 기계의 멈춤 집합과도 같지 않은 새로운 자연수의 집합 D 를 상정할 수 있다. 그리고 우리는 이 집합 D 의 개념을 이용해 튜링 기계를 통해 해결할 수 없는 문제를 만들어 낼 수 있다. 우리는 D 를 대각선 방법에 의해 어떠한 튜링 기계의 멈춤 집합들과도 다르게 만들었기 때문에 “주어진 자연수가 집합 D 에 속하는지 결정하는 알고리듬을 찾아라.”라는 문제는 튜링기계로 해결할 수 없다. 결국 튜링은 튜링 기계를 통해 알고리듬으로 해결할 수 없는 문제가 있다는 것을 증명한 것이다.

3. 결론

대부분의 발명품들이 단 한 사람의 업적만으로 탄생하는 것은 아니다. 많은 사람들이 쌓아 놓은 연구, 생각과 문제들이 마지막 사람에 이르러 집합되는 것이다. 그러나 인류의 발명품 중에서도 컴퓨터는 티끌모아 태산의 종결자라고 볼 수 있다. 왜냐하면 17 세기 라이프니츠부터 시작하여 20 세기 튜링에 이르기까지 오랜 시간 수많은 수학자, 공학자, 논리학자들의 노고가 쌓여 만들어진 것이기 때문이다. 그 중에서도 크게 라이프니츠, 불, 프레게, 칸토어, 힐베르트, 괴델 그리고 마지막으로 튜링이 컴퓨터 탄생에 큰 역할들을 하였다.

컴퓨터의 선조라고 할 수 있는 튜링기계를 만든 튜링에게 직접적으로 영향을 준 사람은 괴델과 힐베르트이다. 힐베르트는 산술의 무모순성을 믿고 이것을 증명하고자 하였다. 그러나 괴델은 산술의 무모순성을 증명하는 과정에서 자신은 이것을 증명할 수 없다는 것을 증명하게 된다. 모든 수학적 명제의 참.거짓을 판단할 수 있을 것이라고 믿은 수학자들에게는 좌절을 안겨준 것이다. 그리고 튜링은 결정 불가능한 명제가 있음을 재증명하는 과정에서 증명의 도구로써

튜링 기계를 사용하게 된 것이다. 이후 튜링기계의 테이프는 메모리칩으로, 테이프를 읽고 쓰는 장치는 메모리칩과 입출력 장치로, 그리고 작동 규칙표는 CPU 로 발전하면서 튜링의 이런 보편 만능 기계는 현대 컴퓨터로 구현되기에 이르렀다.

그러나 괴델과 힐베르트 이외에 라이프니츠, 불, 프레게와 칸토어가 없었으면 컴퓨터가 완성될 수는 없었을 것이다. 라이프니츠는 논리 체계를 계산으로 환원하고 계산을 실행할 엄청난 기계 기관을 꿈꾸었다. 라이프니츠의 생각을 시작으로 그 이후의 학자들의 연구가 펼쳐지게 된다. 불은 자신이 의도하지는 않았지만 최초로 논리학을 대수적인 방법으로 해명하고자 하고 그것을 불 대수라 명명하여 체계화하였다. 그리고 프레게는 인간의 모든 연역적 추론을 그럴듯하게 설명할 수 있는 규칙 체계를 마련하였다. 여기서 힐베르트가 프레게의 규칙들이 그 결론이 전제들에서 유도되게끔 할 수 있는지를 언제나 결정할 수 있게 해주는 명백한 계산 절차를 추구하게 된 것이다.

이렇듯 튜링의 논문은 그 자신만의 것이 아니라 400 년 동안 여러 혁신가들의 노력이 모두 담겨있다. 그러므로 컴퓨터를 제대로 이해하기 위해서는 튜링기계뿐만 아니라 그 안에 녹아있는 라이프니츠부터 괴델까지의 생각, 체계, 증명들을 살펴보는 것이 대단히 유의미한 일인 것이다.

<참고문헌>

마틴 데이비스, 『수학자, 컴퓨터를 만들다』, 박정일・장영태 옮김. 서울: 지식의 풍경, 2005.

2013 년 1 학기 컴퓨터과학이 여는 세계 강의 노트 및 필기 노트