

## 컴퓨터, 그 위대한 “발견”

20-21세기 인류의 방향을 송두리째 바꾼 컴퓨터는 단 한명의 천재로 시작된 것이 아니었다. 논리적인 언어에 반응하는 보편만능의 기계에 대한 꿈은 라이프니츠까지 거슬러 올라간다. 라이프니츠의 꿈은 1850-80년대 부울과 프레게의 논리체계를 통해서야 본격적으로 진행이 되었다. 힐베르트는 당시 최고의 수학자로서 ‘수리명제 자동판별’의 문제를 던졌지만 결국 괴델에 의해 이는 불가능함이 증명되었고, 튜링은 1936년 자신만의 방법으로 괴델의 증명을 확인하고자 ‘튜링기계’라는 개념을 고안해냈다. 이 수업을 통해 들여다본 컴퓨터의 시작은 가까운 미래의 효율성만이 중요한 것은 아님을 깨닫게 했다. 또한, 위대한 업적은 그저 한 개인의 노력으로 이루어지는 것이 아님을 볼 수 있었다. 앞으로도 개인의 노력을 과도하게 찬양할 때, 혹은 필요 이상으로 효율성만을 요구할 때, 그리고 실패에 다다랐을 때, 컴퓨터의 탄생을 다시 기억해야 할 것이다.

20-21세기 인류의 방향을 바꾼 컴퓨터는 단 한명의 천재로 시작된 것이 아니었다. 현재의 컴퓨터는 인간의 삶의 중심에서 일하고 있지만, 그 시작은 너무나도 작았다. 누구도 수학적 꿈의 좌절을 독자적으로 증명하기 위했던 논문이 가져올 파급력을 예상하지 못했다. 컴퓨터의 기원은 단순히 튜링에서 시작되는 이야기도, 튜링에서 멈추는 이야기도 아니다. 컴퓨터의 기원은 오랜 시간에 걸쳐 일궈낸 지식과 그 속에 녹아들어가는 노력의 과정에 있다. 이는 자신의 분야에서 묵묵히 연구를 하며 지식을 축적해 온 모든 학자들, 그들 모두의 이야기이다.

논리적인 언어에 반응하는 보편만능의 기계에 대한 꿈은 라이프니츠까지 거슬러 올라간다. 현재의 컴퓨터는 단순히 주어진 일만 할 수 있는 “계산을 하는 도구”를 넘어서 기계적인 일이라면 무엇이든 해낼 수 있는 보편만능의 기계다. 라이프니츠는 1702년 0과 1만을 사용해 참/거짓 상태를 나타내는 논리적 시스템을 처음으로 구축했다. 그는 연산 작업을 자동화하는 데 흥미를 가졌고, 인간의 모든 사고와 논리구조 또한 기호로 단순화시킬 수 있다고 생각했다. 그의 꿈은 세상의 모든 알고리즘의 참/거짓을 항상 판별할 수 있는 기계를 만드는 것이었다.

라이프니츠의 꿈은 1850-80년대 부울과 프레게의 논리체계를 통해서야 본격적으로 진행이 되었다. 1830년 초반 배비지가 기계적인 연산을 해내고자 해석기관을 만들기 위한 시도를 하기도 했으나 라이프니츠의 꿈을 이루기엔 부족함이 있었다. 그러나 1854년 부울은 0과 1을 사용해 AND, OR, NOT을 표현하는 부울대수를 구축해 사람들의 생각과 말도 논리기호로 표현할 수 있음을 보였고, 프레게는 이 논리체계를 더욱 정교하게 발달시켜 명제의 참/거짓 여부를 수학적으로 찾아내는 과정에 큰 영향을 미쳤다. 이러한 논리학에 대한 연구는 추후에 컴퓨터를 현실화하는데 큰 도움을 주었다. 칸토어의 대각선방법과 무한의 크기에 대한 증명 또한 튜링이 멈춤문제의 해결이 불가능함을 증명하는 데 사용되었다. 이전의 수학자들이 시도하지 않았던 새로운 분야의 탐색이었기에 오직 자신의 직관에 기반을 둔 증명이었다.

힐베르트는 당시 최고의 수학자로서 ‘수리명제 자동판별’의 문제를 던졌지만 결국 괴델에 의해 이는 불가능함이 증명되었다. 힐베르트는 수학적 체계의 무모순성을 증명하려는 과정에서 기호논리학에 입각해 수리명제 자동판별이 가능함을 밝히려고 했고, 이는 논리명제의 타당성을 결정해준다는 점에서 라이프니츠의 꿈과 같은 맥락에 있다. 그러나 괴델은 1931년 불완전성 이론을 내세우며 기계적으로 수학의 참인 명제를 찾아내는 데는 한계가 있다고 주장했다.

힐베르트의 결정문제에 대한 수업을 들은 튜링은 1936년 자신만의 방법으로 괴델의 증명을 확인하고자 몇 가지의 부품을 사용해 일련의 알고리즘을 수행할 수 있고, 인코딩된 프로그램에 따라 무엇이든 할 수 있는 ‘튜링기계’라는 개념을 고안해냈다. 그러나 현재의 컴퓨터는 튜링의 디자인 이후 폰 뉴먼과 새넌과 같은 학자들의 연구가 없었더라면 존재하지 않았을 것이다. 1945년 뉴먼은 EDVAC라는 컴퓨터 구조를 고안해내 하드웨어를 만들어냈고, 새넌은 부울대수가 스위치 원리에도 적용됨을 발견해 현재 모든 컴퓨터의 기본 원리로 사용되고 있다.

이 수업을 통해 들여다본 컴퓨터의 시작은 가까운 미래의 효율성만이 중요한 것은 아님을 깨닫게 했고, 예측의 한계를 보여주었다. 경제학과 진입을 희망하는 학생으로서 효율성을 자주 계산하게 된다. 그러나 컴퓨터의 발명은 응용가능성이 의심스러운 연구들과, 이미 증명된 결과를 재 증명하는 과정에서 만들어졌다는 부분에서 효율성과는 거리가 멀었다. 수학을 논리학으로 환원시키고자 만들었던 논리체계가 컴퓨터의 기원에서 중요하게 쓰이라고는 프레게 자신도 예상치 못했을 확률이 높다. 이처럼 특정 과학적 분야의 연구와 투자가 당장은 낭비라고 생각될 수도 있으나, 그 연구가 미래에 어떤 생각지도 못한 부분에서 중요하게 활용되고 발전될 수 있을지는 아무도 모르는 것이다. 최근 한국은 “주류”연구에만 과학계 치중하는 경향이 있다. 그러나 컴퓨터의 기원이 주는 교훈은 “주류”인 과학적 연구에만 투자하는 것이 아니라, 여러 분야의 기초적 연구가 잘 될 수 있도록 장려해야한다는 것이다. 기초 연구야말로 우리 지식체계의 견고한 기반이 되어줄 것이고, 그럴 때 응용과학의 발달 또한 더욱 빨라질 것이다. 기초과학적 지식이 견고하게 다져져있지 않은 상태에서의 선택적 투자는 당장은 효율적인 것 같아도 불안정할 수밖에 없으며, 지금처럼 시대가 빠르게 변화하는 현대에는 그 흔들림이 훨씬 큰 문제로 다가올 수밖에 없다.

컴퓨터의 발달 과정은 위대한 업적은 그저 한 개인의 노력으로 이루어지는 것이 아님을 보여준다. 오늘날 우리는 필요 이상으로 개인의 노력이 가지는 힘에 열광하고, 모든 실패와 실수는 그 개인의 노력부족 탓으로 돌린다. 하지만 컴퓨터의 발달과정은 꼭 그렇지만은 않음을 알려준다. 컴퓨터의 개념적 디자인과 그 실현은 튜링 이전의 학자들의 노력을 바탕으로 후대의 학자들이 계속해서 연구를 한 결과 나온 열매였다. 컴퓨터의 디자인을 생각해낼 수 있었던 이유는 튜링 자신의 비상한 머리 덕분인 것도 있지만, 그가 그 내용을 공부할 기회가 주어졌기 때문이다. 천재는 혼자 만들어지는 것이 아니라고 했던가. 설령 자신의 엄청난 노력이 함께 했다고 해도, 특정한 교육을 받지 못했거나, 이전의 연구가 되어 있지 않았거나, 기회가 주어지지 않았다면 그렇다할 업적을 남기지 못했을 것이다. 이러한 이유들 때문에 우리는 기초학문의 발달을 장려하고, 최대한 많은 사람들에게 교육의 기회를 제공할 수 있어야한다. 컴퓨터의 시작만 봐도 튜링의 디자인을 위한 기초는 사실 몇 백 년 전부터 준비되고 있었던 것이다. 그렇기에 컴퓨터의 시작은 엄밀히 따지자면 실수도 우연도 아니다. 우연적 발견이라고 주장하기에는 튜링의 논문이 완성되기까지 축적되어야만 했던 수많은 학자들의 새로운 분야의 탐색과 노력, 그리고 성과가 너무 과소평가된다. 그런 의미에서 컴퓨터는 발명된 것이 아니라 “발견”된 것이라고 표현하는 게 더 적절할지도 모른다. 라이프니츠의 꿈부터 현재의 컴퓨터까지, 몇 백 년에 걸친 인류의 협력 작품. 어쩌면 필연적이었던 컴퓨터의 ‘발견’이었다.

앞으로도 우리는 개인의 노력을 과도하게 찬양할 때, 혹은 필요 이상으로 효율성만을 요구할 때, 그리고 실패에 다다랐을 때, 컴퓨터의 탄생을 다시 기억해야 할 것이다. 컴퓨터는 실패를 확인하는 과정에서 “우연히” 발견되었다. 인류의 발전에 있어서 우리는 어떤 태도를 가져야 하는가. 최대한 많은 이들에게 기회를 제공하고, 기초학문의 발전을 장려하며 실패에 다다르더라도 그 실패를 새로운 생각의 기회로 받아들일 때, 우리는 비로소 인간의 한계를 다시 한번 넘어서고, 인류의 역사를 혁신적으로 변화시킬만한 새로운 발견을 할 수 있을 것이다. 그리고 이 때 우리는 몇 백 년 후의 또 다른 튜링을 위한 준비를 하고 있는 것이리라.

## 참고문헌

- 이광근, 『컴퓨터과학이 여는 세계』, 서울:인사이트, 2015, p19-65
- 데이비스, M. 『수학자, 컴퓨터를 만든다』, 박정일(역), 서울:지식의풍경, 2005
- Davis, M. and Ifrah, G. (2002). The Universal Computer: The Road from Leibniz to Turing. *The American Mathematical Monthly*, [online] 109(6), p.581. Available at: <http://www.jstor.org/stable/2695463> [Accessed 22 Sep. 2015].