컴퓨터 세계의 문 앞에 서서

최서현

요약

컴퓨터는 한 순간에 뚝딱 만들어진 게 아니다. 1931년, 수학계의 꿈이 산산조각난다. 튜링이 괴델의 불완전성 정리*incompleteness theorem*를 자기만의 방식으로 다시 증명한다. 살짝 비켜간 사고가 획기적인 발명을 낳았다. 튜링은 멈춤 문제를 푸는 기계가 존재할 수 없다는 증명을 통해 불완전성 정리를 밝힌다.

튜링의 논문에 등장하는 증명들은 흥미로웠다. 하지만 '멈춤 문제를 푸는 기계 H는 존재할 수 없다'는 튜링의 마지막 펀치, 이 펀치는 나에게도 펀치를 날렸다. 튜링기계의 한계를 이야기하는 부분도 어려웠다. 튜링기계를 직접 보고 싶었다. '컴퓨터과학이 여는 세계' 수업을 들을 수 있는 방식이 3가지라는 건 엄청난 복(福)이다.

컴퓨터는 보편만능의 도구다. 내가 살고 있는 세상을 보니 컴퓨터가 없는 곳이 없었다. 컴퓨터를 공부해야겠다는 생각이 들었다. 하지만 순수 예체능계 학생인 내가 컴퓨터를 공부하는 과정은 녹록치 않았다. 남들이 1을 이해하기 위해 1을 공부하면 나는 1을 알기 위해 10을 공부해야 한다. 컴퓨터 공부를 포기하지 않을 거다. 인간은 위대하다.

본문

컴퓨터는 한 순간에 뚝딱 만들어진 게 아니다. 1600년대에 라이프니츠(Gottfried Leibniz, 1646년-1716년)는 심벌을 사용해 논리 추론의 패턴을 정의하려고 시도했다. 이후 프레게(Gottlob Frege, 1848년-1925년), 러셀(Bertrand Russell, 1872년-1970년), 화이트헤드(Alfred Whitehead, 1861년-1947년) 등이 논리적인 추론 과정이 무엇인지 찾아 나갔다. 그런 중 힐베르트(David Hilbert, 1862년-1943년)가 수학의 추론 과정을 담는 기계적인 패턴을 찾는 게 가능할 거라고 제안한다. 이에 괴델(Kurt Gödel, 1906년-1978년)이 반박한다. 이를 튜링(Alan Turing, 1921년-1954년)이 다시 확인한다. 수학계에 청천벽력을 안겨준 논문, 여기서 컴퓨터의 청사진이 드러난다. 400년 전부터 축적된 수학계의 꿈. 그 역사를 공부하고 주류 수학계의 꿈에 의심을 품은 청년에 의해 컴퓨터가 시작한다. 400년 전에 했던 생각이 꼬리에 꼬리를 물고 시대를 흘러와 팡 터진 것이다. 제각각 흩어져있던 사고와 기계장치, 구현 기술이 합쳐져 빛을 발한 게 컴퓨터다.

1931년, 수학계의 꿈이 산산조각난다. 1928년에 당대 수학계를 주도한 힐베르트는 수학자들이 사용하는 추론 규칙을 찾아보자고 제안했다. 그는 기계적인 방식으로 수학의 모든 사실을 자동으로 만들 수 있을 거 같다고 생각했다. 그로부터 3년 뒤 괴델이라는 젊은 수학자가 벼락을 내려친다. 괴델은 기계적인 방법만으론 수학에서의 모든 사실을 만드는 게 불가능하다는 걸 증명한다. 기계적인 방식으로 참과 거짓을 판단할 수 없는 명제가 반드시 존재한다는 거였다.

튜링이 괴델의 불완전성 정리*incompleteness theorem*를 자기만의 방식으로 다시 증명한다. 그는

한 강의에서 괴델의 증명 소식을 듣는다. 그리고 그 증명을 다르게 내보인다. 같은 내용을 증명하는 데 색다른 방식을 얹은 것이다. 튜링은 1936년에 <계산가능한 수에 대해서, 수리명제 자동생성 문제에 응용하면서>라는 논문을 런던 수리학회에 제출한다. 여기엔 젊은 청년의 과감한 시도가 담겨있다. 튜링은 이 논문에서 컴퓨터의 원초적인 도안을 제시한다.

살짝 비켜간 사고가 획기적인 발명을 낳았다. 튜링은 기계적인 방식을 정의한다. 단순한 부품들을 규정하고 그것들로 특별한 기계를 선보인다. 궁극의 기계. 이 기계는 보편만능의 기계 *universal machine*다. 튜링기계의 일종이지만 정한 범주에 드는 모든 기계적인 계산을 하나의 기계로 돌릴 수 있기 때문이다. 여기서 컴퓨터의 주된 기능이 뭔지 짚어보자. 하나의 도구로 모든 일을 할 수 있다는 점이다. 이렇게 튜링이 만든 궁극의 기계는 컴퓨터의 원천 설계도가 된다.

튜링은 멈춤 문제*halting problem*[1]를 푸는 기계가 존재할 수 없다는 증명을 통해 불완전성 정리를 밝힌다. 목표는 '모든 참인 명제를 차례로 만들 수 있는 튜링기계가 존재하지 않는다'는 걸 증명하는 거다. 임의의 튜링기계와 입력을 받아 그 튜링기계의 멈춤 여부를 확인하는 튜링기계가 존재할 수 있을까? 만약 모든 참인 명제를 차례로 만들 수 있는 튜링기계 *A*가 존재한다면 멈춤 문제를 판단하는 튜링기계 *H*를 만들 수 있다. 그러므로 *H*가 존재하지 않으면 *A*가 존재하지 않는다. 튜링은 이 추론을 이용해 *H*가 존재하지 않음을 증명한다. 이로써 모든 참인 명제를 차례로 만들 수 있는 튜링기계 *A*는 있을 수 없다는 것이 입증된다.

튜링의 논문에 등장하는 증명들은 흥미로웠다. 튜링기계의 개수가 자연수의 개수를 넘지 못하는지에 대한 설명부터 멈춤 문제의 증명까지. 역시 수학이라, 답이 명쾌하고 확실하다. 순서를 따라가다 보면 어느새 결론에 고개를 끄덕일 수 있었다.

하지만 '멈춤 문제를 푸는 기계 *H*는 존재할 수 없다'는 튜링의 마지막 펀치, 이 펀치는 나에게도 펀치를 날렸다. 과정이 헷갈렸다. 증명을 이해하는 데 3주가 꼬박 걸렸다. 모든 참인 명제를 차례로 만들 수 있는 튜링기계 *A*가 존재한다면 멈춤 문제를 푸는 그런 *H*를 쉽게 만들 수 있다는 부분이 납득이 안 갔다. 교재에 있는 고작 5줄이 이해가 가지 않았다. 사실 아직도 내가 제대로 파악한 건지 아리송하다. 알면서도 모르겠고 모르면서도 알겠는 부분이 많다. 지금까지 10번 가량 같은 부분만 반복해 읽었다. 읽을 때마다 새로운 곳에서 막힌다.

튜링기계의 한계를 이야기하는 부분도 어려웠다. 튜링기계의 개수는 무한히 많지만 셀 수 있을 만큼만 많다… 자연수의 개수를 넘지 못한다는 말은 알겠다. 그런데 계속 궁금증이 피어올랐다. '테이프에 표현된 튜링기계를 고유의 자연수 하나로 표현할 수 있다는 게 뭐지?'[2]라는 기초적인 질문에서부터 시작해 '심벌들은 왜 17개지?', '17진법과 10진법을 같이 사용해 변환하는 건 왜지?'하는 물음도 가졌다. 이런 의문들이 연달아 들면 '나는 왜 결론은 알겠는데 과정을 모르겠지?'하는 반성적인 생각이 든다.

튜링기계를 직접 보고 싶었다. 기계라면 내 눈 앞에서 움직이는 게 보여야 맛깔나는 법이다. 나는 시뮬레이션을 원했다. 그림으로만 보니 이해가지 않는 부분이 더 많았다. 특히 여러 개의 테이프를 한 테이프에 합치는 부분은 납득하기가 힘들었다. 튜링이 실제로 기계를 만들었는지 고안만 했는지는 모르겠다. 그렇지만 튜링이 아니더라도 누군가 튜링기계를 시뮬레이

---

1) 어떤 튜링기계가 멈출지 안 멈출지를 정확히 판단하는 문제

2) 지금은 이 의문을 해소했다. 처음엔 기계적 장치로만 배운 튜링기계를 숫자로 표현한다는 게 어떤 모양을 가지는지 궁금했다. 사람에 출석번호를 붙이는 거로 비유하니 납득이 갔다. 이토록 쉽게 이해할 수 있는 사실에 대해서 의문을 가진 내가 한심했다.

션 해놓은 자료가 있을 거 같았다. 그래서 찾아봤다. 유튜브에 ‘Turing machine'을 검색했다. 나 같은 사람이 많았는지 'Turing machine explained', 'Turing machine example' 등이 자동검색으로 떴다. 다행이었다. 하나의 시뮬레이션만 나와도 한숨 돌렸을 텐데 꽤 많은 영상이 나왔다.

‘컴퓨터과학이 여는 세계’ 수업을 들을 수 있는 방식이 3가지라는 건 엄청난 복(福)이다. 수업을 듣고 모르는 게 있으면 먼저 책이나 인터넷 강좌로 그 부분을 공부할 수 있다. 그래도 모르는 게 있으면 질문하면 된다. 책이나 인터넷 강좌가 없었다면? 수업을 포기했을 수도 있다. 컴퓨터과학은 처음 공부하는 분야다. 처음 공부하는 데 한번 듣고 이해하는 사람은 없을 거다. 나는 그렇다. 반복해서 공부할 수 있는 삼박자가 갖춰진 이 시점에 수강을 한 건 신의 한 수다.

컴퓨터는 보편만능의 도구다. 얼마 전 ‘인류학의 이해’ 수업 시간에 구석기의 만능도구에 대해 배웠다. 교수님이 “만능도구란 어떤 거예요?”라는 질문을 던졌다. 그 질문에 대한 답은 이거였다. “만능도구는 좋은 게 아니에요. 만능은 어느 하나도 제대로 못한다는 거예요.” 나는 컴퓨터를 예시로 반박하고 싶었지만 참았다. 컴퓨터란 도구는 다재다능하다. 소통도 할 수 있고 영화도 볼 수 있고 쇼핑도 할 수 있다. 심지어 이 모두를 완벽하게 해낸다. 컴퓨터로 할 수 없는 걸 찾는 게 난관이다. ‘운동은 못하네!’하고 생각했더니 몸으로 하는 게임[3]도 있지 않나.

내가 살고 있는 세상을 보니 컴퓨터가 없는 곳이 없었다. 내 수중에 있는 노트북은 물론이고 지하철에 있는 전광판, E-BOOK, 강의실의 프로젝터, 초인종, 세그웨이[4] 등. 컴퓨터는 앞으로 더 발전할 것이다. 벌써 로봇과 인공지능이 대두하고 있다. 현재 전통 학문으로 인정받는 철학, 의학, 예술 등의 역사와 비교해보자. 앞서 말한 학문들의 역사는 몇 백만 년이다. 이에 비하면 컴퓨터는 역사가 100년 채 되지 않은 젊은 발명품이다. 그런데 이 놀라운 물건이 없는 분야를 상상할 수 있는가. 병원에서 환자들의 차트를 효율적으로 구축할 수 있던 건 컴퓨터 덕이다. 수술실에선 로봇이 보조하는 수술이 행해진다. 예술에도 기술이 들어서고 있다. 경영, 교육, … 어느 분야든 마찬가지다.

컴퓨터를 공부해야겠다는 생각이 들었다. 급속히 발전하는 컴퓨터 기술에 둘러싸여 살면서 이걸 배우지 않으면 휩쓸려 내려갈 수 있겠다는 불안함이 엄습했다. 냇물이 폭포를 만난 거처럼. 강렬한 물길에 걸음을 주체하지 못하고 넘어질 수 있겠단 마음이었다. 이 세상에서 주도적인 삶을 이끌려면 컴퓨터과학을 배우는 게 필수인 거 같다. 내가 뭘 하든 컴퓨터를 접할 테니까. 100년도 되지 않았는데 이토록 빠른 발전을 이룬 컴퓨터. ‘더 발전할 게 있나?’하는 의구심은 안일했다. 아직 컴퓨터는 아주 조금 발전한 거라고 한다. 그만큼 발전가능성이 광범위하다. 디지털 세계에 사는 나. 컴퓨터를 공부하는 건 세상에 대한 예의인 거 같다.

하지만 순수 예체능계 학생인 내가 컴퓨터를 공부하는 과정은 녹록치 않았다. 수학을 놓은 지 거의 4년째다. 수업을 듣는데 논리적인 사고가 계속 중단됐다. 4년 동안 굳어있던 수학 머리를 가동하려니 보통 힘든 게 아니었다. 하나를 이해하지 못하는 상태로 둘로 넘어가고 셋으로 넘어가고 …….‘이걸 계속 공부하는 게 맞는 걸까?’하는 회의감이 들었다. 교수님은 고등학교 때 배운 수학이면 충분히 이 내용을 이해할 수 있다고 하셨다. 그런데 난 고등학교 2학

3) 예를 들어 XBOX360 KINECT나 닌텐도 WII. 이런 게임 소프트웨어도 결국 컴퓨터에서 시작한 것이다. 실로 컴퓨터는 인간의 놀이를 확장시킨다.

4) 아이봇(IBOT)의 발명가 딘 카멘Dean Kamen이 발명한 1인용 스쿠터다.

년 때부터 수학을 하지 않았다. 컴퓨터과학을 공부하기 위해 다시 수학을 공부해야 할까? 컴퓨터공학의 수학은 다른 공학의 수학과 다르다고 하는데 그럼 난 어떤 수학을 공부해야 할까? 포기해야 할까? 끊임없이 질문하고 의심을 품었다.

남들이 1을 이해하기 위해 1을 공부하면 나는 1을 알기 위해 10을 공부해야 한다. 10에 근접하게 공부했을 때야 수업을 따라갈 수 있었다. 같은 부분의 강의를 3~4번 돌려보고 책을 5~7번씩 훑었다. 이젠 책의 앞부분이 말려 올라간다. 그렇게 해서야 비로소 깨달았다. 복습을 안 하면 다음 수업 시간엔 더욱 힘들었다. 내가 선택한 공부인데 그거로 스트레스를 받고 싶지 않았다. 답답하고 회의감이 들 때 마다 나를 위로했다. 원래 처음 배우는 게 쉬울 리가 없잖아! 이렇게 배우는 것들은 나에게 굉장한 자양분이 될 거야.

컴퓨터 공부를 포기하지 않을 거다. 이 세상에서 컴퓨터를 공부하는 게 필수적인 거 같다는 생각. 이 생각이 맞는 건지 무리수인지는 모르겠다. 하지만 이번 학기가 끝나고 내려놓는 한이 있더라도 배우는 동안만큼은 열의를 갖고 공부하겠다.

인간은 위대하다. 수업에서 들은 얘기 중에 인상 깊은 말이 있다. 한 사람의 뇌에는 약 10의 18승개의 스위치가 있다는 거였다. 뇌 하나에 1000억 개의 뉴런이 있고 뉴런들의 연결지점마다 수천 개의 스위치가 있다고 한다. 컴퓨터는 인간 사회를 획기적으로 전환했다. 그런데 이 보편만능의 도구 역시 인간이 개발한 것이다. 사람의 두뇌에서 나온 생각이다. 언젠가 또 다른 누군가가 지금의 컴퓨터를 능가하는 컴퓨터를 생각할지 누가 알까. 컴퓨터를 갓 배우는 사람으로서, 컴퓨터 세계의 문 앞에 서서, 미래엔 어떤 컴퓨터 기술이 내 마음을 두근거리게 할지 기대한다.