**말할 수 있는 것과 없는 것, 개발**

지난 10년간 꾸준히 창작을 해왔다. 시, 소설, 에세이, 가사와 음악 등 생각을 드러낼 수단이라면 가리지 않았다. 즐거웠다. 창작은 밤을 새는 일, 머리를 싸매는 일, 자의식에 허덕대며 방탕을 일삼는 일, 객혈을 하면서도 음주와 흡연을 지속함 등, 모든 고통과 어리석음 마저 즐거움으로 탈바꿈 시켰다. 게으른 적은 있었으나 배움에 대한 관심이 게으른 적은 없었다. 돌아보면 많이 배웠다고 생각한다. 제도권의 교육을 받지 않았으니 남들보다 더 열심히 살아야 한다는 생각이 있었고, 다른 식으로 공부해야 한다는 열망이 컸다. 그래서 독서를 많이 했다. 지금에야 세상에는 다독으로 해결할 수 없는 전문 지식과 테크닉이 있다는 사실을 잘 안다. 당시에는 그런 앎이 어렴풋했다. 그래서인지 내 방향은 깊이를 쌓아가는 학습과는 거리가 멀었다.

불과 삼, 사 년 전까지 나는 철학이 세상의 근본이라 믿었고 그를 위해 공부했다. 상대성을 포용하는 철학 답게 나는 서로 다른 사상을 넓고 얇게, 때로는 맥락을 고려하지 않고도 깊게, 자의적으로 해석해가며 자의식을 키워왔다. 나는 한결 더 복잡한 사람이 되었다. 철학과 문학을 주류로 얕은 인류학, 역사, 과학을 곁들여 세상의 사소한 문제를 더 큰 문제로 부각하는 창조 활동에 몰두하고는 했다. 지금보다 훨씬 자의식이 강하던 시절의 나는 당시의 내 생각에 몰입해 주변을 잊곤 했다. 내가 생각하는 문제가 세상에서 가장 중요하고 해결이 시급한 문제라는 생각이 있었다. 당시에는 정말 그 문제를 위해 투신하겠다는 각오를 다지며 살아갔으므로, 얼마간은 사실이었을지도 모르겠다. 또 내 평생의 독자는 나를 알아보는 사람 뿐이라는 독선도 잊지 않았다. 나를 통해 나온 무엇은 그런 사람들의 창조와 혁신에 불을 지르고 사라지는 방화범 내지는 불쏘시개였으면 했다. 불쏘시개만큼 하찮아도 그 불꽃에 영향을 미칠 수만 있다면 상관없었다.

그러나 세상 문제의 팔 할이 그렇듯, 어디까지나 돈이 문제다. 좋은 사람들 덕에 나는 세상과의 접점을 유지한 채 속물성을 잊지 않을 수 있었다. 속물성이 없었더라면 지금의 나는 부코스키의 시에 나온 변변치 않은 잡놈이 되어 주변인들의 피를 축냈을 것이다. 돈에 대한 문제의식이 없으면서, 막상 돈 없이는 살지 못하는 함량 미달의, 타칭 예술가의 삶은 얼마나 처참한가? 사람들은 그를 향해 말한다. 저 사람은 가난한 삼류 예술가라고, 하지만 그는 생각한다. 자신은 아직 발굴되지 않았을 뿐이라고. 자신이 하는 일은 예술이라는 고매한 틀을 쓰기에는 너무 변변치 않다고, 하지만 그 변변치 않음으로 그의 행위가 변혁에 속하는 일이라고. 그리고 아무도 그가 그런 생각을 하는지 모른다. 스스로 아무리 떠들고 궤변하고 연설을 해도 사람들은 그가 가난한, 함량 미달의 삼류 예술가라고 생각한다. 그리고 그에 대한 주변인들의 그러한 생각은 팔 할이 정답이다.

사실 이렇게 말하면서도 나는 그렇게 믿지 않는다. 나에 대한 나의 앎과 기대와 주변인들의 앎에는 매워지지 않는 간극이 있다. 그렇기에 둘러대기 위해 하는 자아비판과 같은 것이다. 여기에는 다시 참에 해당하는 팔 할의 진리값과 거짓에 해당하는 이 할의 진리값이 있다. 말할 수 없는 것에 대해 이야기하기 위해서는 이런 형태의 Fabrication이 필요하다. 프로그래밍의 추상화와 크게 다르지 않다.

부분에 대해서만 말하고 더 큰 부분과 세부사항에 대해 말하지 않음으로 가려진 부분이 드러나기도 한다. 내가 어쩌다 프로그래밍을 시작하게 되었는가에 대한 설명이 되었으리라 생각한다. 곡해의 여지를 줄이기 위해 부연하자면 순전히 돈을 벌기 위해 시작한 것 만은 아니다. 프로그래밍은 창작과 돈을 벌어야 함의 문제를 동시에 해결하는 이상적인 직업이기도 했다. 더군다나 파고들수록 배울 것이 많은 깊이와 복잡성에 매료되기도 했다. 정보를 다루며 개념을 현실로 구현하는 작업이 세상의 중요한 부분을 건드린다는 생각을 갖게 됐고 지금은 내 관심이 향하는 거의 전부가 되었다. 그러나 내가 지나온 시간에서 배워온 강점을 살릴 수 있는 분야라는 초기 생각과 달리 프로그래밍은 글쓰기와 아주 많은 차이가 있었다. 더 자세히 말하자면 논리와 명료함을 멀리하고 논리와 논리 사이의 간극을 그려내고 싶어하던 문학적 글쓰기 방식과 프로그래밍에는 근본적인 차이가 있었다.

넓게 흐르는 강의 양 옆으로 두 개의 마을이 있다고 가정한다. 사람들은 두 마을에 대해 궁금해한다. 문학적 글쓰기는 흐르는 강의 깊이와 유속에 관해, 한낮의 해가 물결 위로 부숴지는 일에 관해, 강의 시작과 끝에 무엇이 있는지, 강이 끼고 도는 마을의 풍경에 관해, 흘러온 강의 역사와 강을 사이에 두고 벌어진 사람들의 갈등과 꿈에 관해 이야기한다. 그러나 프로그래밍은 다리를 지어 두 마을을 연결한다. 프로그래밍의 목적은 최대한 정교하고 튼튼한 다리를 지어 오래토록 유지시키고 필요하다면 용도에 맞게, 항구가 들어서기 편하도록, 뗏목을 연결하는 줄을 달 수 있도록 준비해두는 일이다. 달리 말해 문학적 글쓰기는 다리를 놓는 일에 오히려 방해가 되고는 했다.

따라서 프로그래밍은 명시지의 향연이다. 프로그래밍에서는 추상화 하려는 그 대상조차 명시지를 향해 있음을 주지해야 한다. 사변성과 팔 할의 진실, 이 할의 거짓 따위의 모호성은 버그를 낳는다. 결국 개발은 온전히 말할 수 있는 것에 대해서만 말해야 한다. 한 단계 더 나아가 말할 수 있는 것을 향해 말할 수 있는 것만 말함으로써 말할 수 없는 것을 말하게 되는 세련된 일이기도 하다.

결론적으로 내가 해온 일이 진정으로 암묵지를 쌓고 암묵지로서 기능하는 일들이었다면 굳이 컴퓨터 공학을 공부하면서까지 그런 일들을 할 필요가 없다. 프로그래밍은 드러내고 보여줄 수단으로 기능해야지 이전처럼 복잡성을 증대시키는 일이 되어서는 안 된다. 따라서 보다 실용적이고 문제해결에 가까운 일부터 학습해야 한다. 익숙하게 해오던 글쓰기와의 차이점을 구분하고 내가 쌓아온 강점과 약점을 활용할 줄 알아야 한다. 내가 해온 일이 진정으로 의미있는 일이었다면, 그리고 그것이 쌓여 내 강점이 되었다면, 더이상 그것에 오랜 시간을 쏟지 않아도 드러나리라고 생각한다. 명료함을 잃지 말자. 단순화하자. 버리는 연습을 하자.