

이대현 (한국산업기술대학교 게임공학부)

1. 서론

컴퓨터와 프로그래밍을 처음 접한 지 어느덧 20년이 넘게 흘렀다. 당시 컴퓨터의 중요성에 대한 선견지명을 갖고 계시던 아버지께서 30만원에 이르는 거금을 선뜻 내주셔서, FC-80이라는 금성사의 MSX 컴퓨터를 산 첫날의 흥분이 아직도 기억난다. 4MHz의 Z80-A CPU와 32KB/64KB의 ROM/RAM을 지닌, 지금의 기준으로 보면 턱없이 낮은 사양의 컴퓨터에 불과했지만, 그 당시의 다른 컴퓨터들에 비해 월등한 수준의 환상적인 16 컬러 그래픽과 사운드 덕분에 고수준(?)의 게임들을 실했을 수 있었다.

게임에 한참 빠진 후, 자연스럽게 게임의 구현에 대한 공학적인 흥미가 생겼으며, 이는 프로그래밍을 시작할 수 있는 계기가 되었다. FC-80은 꽤 괜찮은 BASIC 인터프리터를 내장하고 있어, 이를 이용해서 직접 코딩을 하고 게임을 만들면서 독학으로 프로그래밍을 배웠다. 일본 회사들의 상용 게임에 비해 형편없는 자작 게임의 성능을 높여보고자, 어셈블리 프로그래밍까지 도전해서 나름대로 성능 최적화를 위해 노력하는 과정에서 컴퓨터의 구조를 이해하게 되었다.

돌이켜보면, 프로그래밍의 첫 시작을 인터프리터 방식의 언어인 BASIC으로 하게 된 점이 프로그래밍을 쉽고 흥미있게 공부할 수 있도록 해 준 것이 아닌가라는 생각을 해본다.

국내 대부분의 컴퓨터 공학 또는 게임 관련 학과들은 1학년 학생들의 프로그래밍 입문언어로 C 또는 C++을 가르치고 있다. 학교에서 10년 가까이 프로그래밍 강의와 게임개발강의를 하면서, 많은 학생들이 C/C++ 프로그래밍 과목을 어려워하고, 결국 프로그래밍 자체에 흥미를 잃게 되고 마는 경우들을 숏하게 보아왔다. 그 원인에 대한 여러 가지의 분석이 있을 수 있겠으나, 20년 전에 프로그래밍을 접한 경험과 기억에 근거하자면, C/C++이 문법적으로 상당히 어려울 뿐만 아니라, 컴파일러 방식의 프로그래밍이 초보자에게는 적합하지 않다는 점에 주목할 필요가 있다.

필자는 지난 8년간, 게임공학부 2학년 학생들을 대상으로 “2D 게임 프로그래밍” 과목을 강의하면

서 실습을 위해 C/C++을 사용해 왔으나, 금년 2학기부터 C/C++ 대신 인터프리터 방식의 언어인 파이썬을 활용하고 있다. 이는 필자가 아는 범위에서는 대학교 정규 게임 개발 관련 교과에 파이썬을 사용하는 국내 최초의 시도이다. 그 결과 파이썬을 활용한 프로그래밍 및 게임 개발 교육의 효과성을 어느 정도 확인할 수 있었다.

본고에서는 효과적인 게임 프로그래밍 교육의 도구로써 파이썬이 갖는 가치와 가능성을 소개하고자 한다.

2. 프로그래밍 입문 언어의 변화

전 세계적으로 소프트웨어와 프로그래밍의 중요성이 부각되면서 몇가지 변화들이 눈에 띈다. 먼저, 프로그래밍의 초기 교육 정책이 미국, 영국 등을 중심으로 활발히 실행되고 있다. MIT의 스크래치(Scratch)는 어린이들을 대상으로 한 프로그래밍 교육을 위해 개발된 도구로써, 프로그래밍의 기본 요소를 도형화한 블록들을 조합함으로써 프로그램을 짤 수 있어, 배우기가 매우 쉽고 직관적이다. 최근에는 국내에서도 초등학생을 대상으로 스크래치를 활용한 프로그래밍 교육 프로그램들이 대폭 늘어나고 있다[1].



그림 1 | 스크래치 실행 화면

프로그래밍 초기 교육에서 중시되는 것은 프로그래밍이라는 주제를 쉽고 흥미있게 접할 수 있게 하자는 것이며, 이를 위한 다양한 방법들이 시도되고 있다.

이러한 맥락에서 파이썬의 가치가 최근 인정을 받고 있는 조사가 발표되었다. 미국 내 주요 대학교의 컴퓨터과학 관련 학과들에서 프로그래밍 입문을 위해 사용하는 언어로 파이썬이 1위를 차지한 것이다[2].

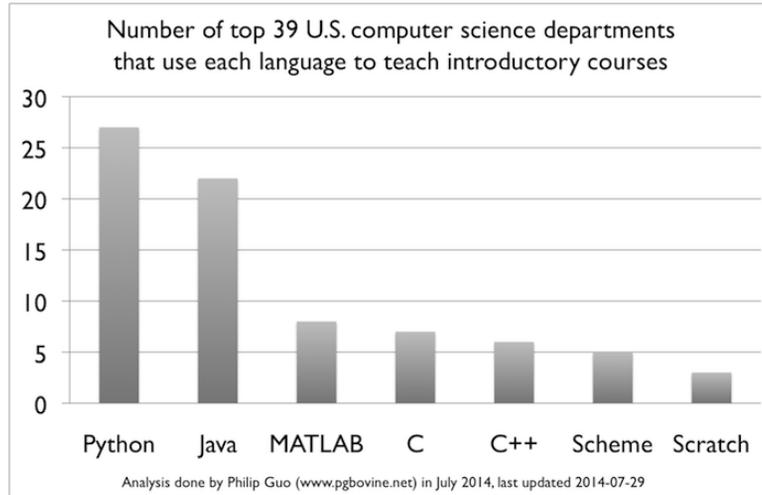


그림 2 | 미국 주요 대학 프로그래밍 입문 언어 사용 통계

이 조사 결과는 교육 현장에서 프로그래밍 교육의 도구로써 파이썬의 효용성을 잘 보여주고 있다. 특히, 파이썬 입문 서적 “Think Python”의 저자는, JAVA를 사용한 프로그래밍 수업과 비교해서 파이썬을 사용했을 때 그 결과가 충격적이라고까지 언급하며, 학생들이 덜 고생하고, 더 많이 배우고, 더 많은 프로젝트를 더 즐겁게 수행했다고 밝혔다[3].

우리나라의 경우, 서울대와 KAIST에서 프로그래밍 입문 수업에 파이썬을 사용한 일부 사례가 있으나, 아직 대부분의 학교에서는 C/C++ 또는 JAVA를 프로그래밍 입문 교육에 사용하고 있는 것이 현실이다.

3. 파이썬 언어의 장점

파이썬은 1991년 귀도 반 로섬(Guido Van Rossum)이라는 프로그래머에 의해 개발된 고급 프로그래밍 언어로, 인터프리터 방식을 채택하고 있으며, 동적 자료형을 특징으로 하는 객체 지향 언어이다. 현재는 비영리 재단인 파이썬 소프트웨어 파운데이션(Python Software Foundation)에서 파이썬 언어의 공식적인 표준화, 개발, 관리 및 배포를 담당하고 있다.

20여년의 짧은(?) 역사에도 불구하고, 파이썬은 강력한 기능과 쉬운 사용성 덕분에 계속 사용자 층을 넓혀왔으며, 2014년 기준 전세계에서 7번째로 많이 사용되는 언어로 부상했다. 특히, 구글은

파이썬을 주요한 프로그래밍 언어로 사용하고 있는 것으로 알려져 있다[4]. 한편, 최근 주목받고 있는 기술 중의 하나인 빅데이터 기술의 경우, 그 구현에 파이썬이 널리 사용되고 있다.

Programming Language	2014	2009	2004	1999	1994	1989
C	1	2	2	1	1	1
Java	2	1	1	3	-	-
Objective-C	3	27	36	-	-	-
C++	4	3	3	2	2	2
C#	5	5	7	18	-	-
PHP	6	4	5	-	-	-
Python	7	6	6	23	21	-
JavaScript	8	8	9	15	-	-

[그림 3] | 프로그래밍 언어 사용 순위(출처: www.tiobe.com)

파이썬이 그 입지를 넓힐 수 있었던 것은 파이썬이 제공하는 다양하고 강력한 기능과 특성덕분이다. 필자가 파이썬을 접하고 익히고 사용하면서 느꼈던 파이썬의 장점은 다음과 같다.

첫째, 파이썬으로 작성된 코드는 매우 읽기 쉽다. 프로그래밍 강의를 하면서 필자가 학생들에게 제일 강조하는 것 중의 하나는 읽기 쉬운 코드를 짜라는 것이다. 가독성의 중요함 때문에 항상 강조하고 있으나, 사실 C/C++로 코드작성을 하다보면, 개발 초기에는 나름대로 가독성을 지키기 위해 노력하지만, 코드가 커지면서 가독성을 유지하는 것이 여간 힘든 게 아니다. 이에 비해 파이썬은 문법적 요소들의 간결함과 잘 준비된 라이브러리 덕택에 우아한 코드를 일관되게 작성할 수 있다.

[그림 4]는 간단한 파이썬 코드이다. 컴퓨터 언어를 조금이라도 알고 있다면, 이 코드를 보면 대충 어떤 일을 하는 코드인지 쉽게 파악할 수 있을 것이다. 이처럼 영어 문장을 읽어내려 가듯이 코드를 읽고 해석할 수 있다는 점이 매우 매력적이다.

```
# simple.py
languages = ['python', 'perl', 'c', 'java']

for lang in languages:
    if lang in ['python', 'perl']:
        print("%6s need interpreter" % lang)
    elif lang in ['c', 'java']:
        print("%6s need compiler" % lang)
    else:
        print("should not reach here")
```

[그림 4] | 파이썬 코드 예 (출처: 점푸투파이썬 <https://wikidocs.net/6>)

둘째, 파이썬을 사용하면, 원하는 기능을 쉽고 빠르게 구현할 수 있다. 사실 프로그래밍 언어를 통해 개발자가 원하는 기능을 본격적으로 구현하려면, 문법적인 것 이외에 복잡한 라이브러리 함수의 사용법에 대한 이해가 필요하며, 이는 초기에 프로그래밍을 어렵게 만드는 요인이 된다. 파이썬은 “battery included”라는 철학 아래, 강력하고 사용하기 쉬운 라이브러리를 기본 제공하고 있어, 초보자들도 쉽게 고급 기능을 구현할 수 있다. 성능 최적화를 위해, 어떤 특정 함수(예를 들어 core()라는 함수)의 실행속도, 호출횟수 등의 프로파일 정보 획득이 필요하다고 가정해 보자. 다른 언어의 경우, 복잡한 프로파일링 준비 작업을 해야 하나 파이썬의 경우, “import profile”로 필요한 라이브러리를 임포트하고, 프로파일링을 원하는 함수에 대해서, “profile.run('core()’)와 같이 호출해 주면 그걸로 끝이다. 얼마나 간결한가?

HTTP 서버를 만드는 것도 파이썬에서는 너무나 간단하다. 아래 그림과 같이 몇 줄 안 되는 코드로 쉽게 HTTP 서버를 만들고 실행할 수 있다.

```
import http.server
import socketserver

PORT = 8000

Handler = http.server.SimpleHTTPRequestHandler

httpd = socketserver.TCPServer(("", PORT), Handler)

print("serving at port", PORT)
httpd.serve_forever()
```

| 그림 5 | Http 서버를 구현하는 파이썬 코드

셋째, 파이썬은 배우기가 매우 쉽다. 프로그래밍에 대한 기본 지식을 갖고 있긴 했지만, 필자의 경우 파이썬의 기본 문법을 습득하고 프로그래밍을 시작하기에 하루면 충분했다. 학습이 매우 쉽기 때문에, 심지어 “파이썬을 10분에 배우기”라는 도발적인 제목의 웹사이트 까지 있을 정도이다[5]. 이번 학기 “2D 게임 프로그래밍” 과목에서 파이썬에 대한 기본 설명은 두시간 정도의 강의로 충분했으며, 그 이후 실제로 게임 개발 실습을 진행함에 전혀 문제가 없었다.

마지막으로, 파이썬은 똑똑한 문법적인 기능들이 많다. 프로그램 개발에 있어 필요한 대표적인 자료 구조인 리스트와 맵 등이 문법적인 요소로 아예 포함되어 있어 그 사용이 무척 쉽다. 또, 기본적으로 자료형에 대한 구별을 따로 하지 않기 때문에, 다형성이라는 어려운 단어를 몰라도, 자연스럽게 다형성을 지닌 코드를 작성할 수 있다. 함수 리턴에 있어 여러 개의 값을 동시에 리턴할 수 있기 때문에, 함수를 다양하게 활용할 수 있으므로 무척 편리하다.

4. 파이썬을 활용한 게임 개발 교육 사례

필자는 한국산업기술대학교 게임공학부 학생을 대상으로 2학년 2학기에 개설되는 “2D 게임 프로그래밍” 과목에 대한 강의를 2005년부터 10년 동안 담당하고 있다. 2005년에는 DirectX 라이브러리와 C++을 이용한 게임 개발을 강의했고, 2006년부터 2013년까지는 SDL 이라는 크로스플랫폼 2D 라이브러리와 C++을 이용한 게임 개발을 강의했다. 2D 렌더링과 2D 물리에 기초한 게임 개발이 중심이긴 하지만, 학생들이 게임 개발 프로젝트를 통해 처음으로 게임다운 게임을 직접 제작해보는 과목이기 때문에, 객체 지향적 관점에서 게임이라는 소프트웨어를 효과적인 개발하기 위한 프레임 워크 등의 내용도 심도있게 다루고 있다. 학생들은 개인별로 프로젝트를 수행해서 완성된 형태의 게임을 제작해야 하고, 2010년 2학기부터는 그 결과를 홈페이지를 통해 공개하도록 되어있다.

발표자	1차 발표	2차 발표	3차 발표	최종 발표
홍길동	1차발표가이드 동영상	동영상 소스코드	동영상 소스코드	동영상 소스코드
윤여민	발표자료 동영상	2차발표 동영상	3차발표 동영상	소스코드 동영상
백태민	발표자료 동영상			
김준호	발표자료 동영상	소스코드 동영상	소스코드 동영상	소스코드 동영상
조재훈	발표자료 동영상	동영상 소스코드	동영상 소스코드	동영상 소스코드
고영준	발표자료(수정) 동영상	소스코드 동영상		
이승준	발표자료 동영상			
이영준	발표자료 동영상	소스코드 동영상	소스코드 동영상	동영상 소스코드
정승윤	발표자료 동영상	소스코드 동영상	소스코드 동영상	소스코드 동영상
최정우	발표자료 동영상	소스코드 동영상	소스코드 동영상	소스코드 동영상
김원성	발표자료 동영상			
노정호	발표자료 동영상	소스코드 동영상	발표자료 동영상	
박근혁	발표자료 동영상	발표자료 동영상	동영상 소스코드	동영상 소스코드
안수근	발표자료 동영상	동영상 소스코드	동영상 소스코드	
홍종대	발표자료 동영상	동영상 소스코드	동영상 소스코드	동영상 소스코드

그림 6 | “2D 게임 프로그래밍” 교과 프로젝트 (출처: <http://daehyunlee.actimedia.co.kr>)

올해 2014년 2학기부터는 과목의 기본 내용은 유지한 채, 실습과 프로젝트에 사용하는 언어를 파이썬으로 변경했다. 학생들이 비록 1학년 C 프로그래밍 수업을 통해 프로그래밍에 대한 기초적인 지식은 갖고 있긴 했지만, 파이썬이라는 언어를 완전히 처음 접하고 사용해야 하는 것이라 쉽지 않은 결정이었다. 파이썬 도입을 결정한 이후, 한참 고민했던 부분은 파이썬 언언 자체에 대한 강의 비중이었다. 파이썬 언어 자체의 강의 비중을 높이게 되면, 2D 게임 프로그래밍 과목에서 다루어야 할 고유의 주제들에 대한 강의를 소홀해질 수 밖에 없기 때문이었다. 고심 끝에, 파이썬의 학습 용이성을 믿고, 파이썬 문법에 대한 강의는 학기 시작 직후 단 두시간만 진행했고, 그 이후 실습을 통해

학생들이 파이썬에 대해 익숙해질 수 있도록 했다.

한 학기 동안 파이썬을 활용하여 게임 개발 강의를 진행하면서, 두가지 관점에서 도입 결과의 가치를 파악하고자 했다. 첫째로, 과연 파이썬이라는 언어가 널리 알려진 바와 같이 학습하기 매우 쉬운가 하는 관점이었다. 이에 대한 결론은 의외로 싱겁게 나왔다. 개강 후 첫 두 시간의 문법 설명 이후, 학생들은 실습을 진행하면서 프로젝트를 수행하는데 그다지 어려움을 겪지 않았다. 종강 후, 강의 소감에서 학생들이 공통적으로 얘기했던 것은, 수업 초기에는 과연 생소한 파이썬을 갖고 프로젝트를 수행할 수 있을까라는 막연한 두려움이 있었으나, 단기간 안에 쉽게 파이썬을 배우고 활용할 수 있게 됐다는 점이다.

둘째, 파이썬이 프로젝트의 수행을 양적 및 질적으로 개선시키는데 영향을 미치는가 하는 관점이었다. 작은 게임이라 하더라도, 학생들이 게임을 처음부터 개발해서 끝까지 완성된 결과물을 내는 것은 결코 쉽지 않은 일이라, 지난 몇 년 간 과목을 수강했던 상당수의 학생들이 프로젝트의 최종 결과물을 제출하지 못했던 것이 사실이다. 프로젝트 완료율의 경우, 필자가 문화체육관광부에서 PD로 2년 동안 파견 근무한 기간을 제외한 지난 4년간의 통계를 보면, 2009년부터 2013년까지의 3년 동안 총 109명의 학생들이 수강했고, 이중 67명의 학생이 최종적으로 프로젝트를 완료함으로써 누적결과로는 61.5%의 완료율을 보였다. 금년 2학기 강의는 총 82명의 학생이 수강해서, 60명이 완료함으로써 73.2%의 완료율을 보였다. 통계학적으로 의미있는 자료가 도출되기까지는 앞으로 2,3년의 추이를 좀 더 지켜볼 필요는 있으나, 프로젝트 완료율의 실질적인 향상을 가져온 것만큼은 분명한 것으로 보인다.

더 중요한 것은 프로젝트의 질적인 수준의 향상이 이루어졌다는 점이다. 상당한 수준의 게임 프로젝트들이 눈에 띄게 증가했고, 전반적으로 완성된 수준이 꽤 향상됨을 실감할 수 있었다.

표 1 | 2D 프로그래밍 과목 프로젝트 완료 현황

년도	총학생수	프로젝트완료 학생수	완료율
2009년	27	19	70.4
2010년	48	22	45.8
2013년	34	26	76.5
2014년	82	60	73.2

한편, 몇몇 학생들은 프로젝트 소감을 말하면서, 프로그래밍을 거의 포기했었는데 파이썬을 통해 프로그래밍에 대한 흥미가 높아졌고, 또 결과적으로 프로젝트를 완료하는 경험을 통해 학습에 대한 자신감이 붙게 되는 계기가 되었다고 밝혔다. 이것은 향후 프로그래밍 입문 교육의 변화에 대한 필요성을 보여주는 사례가 할 수 있겠다.

5. 결론

단어에 “부드럽다”라는 뜻이 들어가 있음에도 불구하고, 소프트웨어를 개발하는 개발자들은 사실 굉장히 “딱딱해서” -보수적이어서-, 주로 사용하는 컴퓨터 언어를 바꾸거나 새로운 언어에 대한 공부를 잘 하지 않는 것으로 알려져 있다. 필자의 경우도 대학교 졸업 이후로 거의 10년이상 C/C++로 대부분의 개발 작업을 했고 강의도 진행해 왔으며 다른 컴퓨터 언어에는 그다지 관심을 기울이지 않은게 사실이다. 그런데, 작년 가을께, 파이썬을 접하게 되었고, 그 매력에 흠뻑 빠지면서 지금은 주로 많은 작업을 파이썬을 이용해서 처리하고 있다.

금년 2학기 2D 게임 프로그래밍 과목의 언어를 파이썬으로 변경하면서, 강의 준비에 상당한 시간을 쏟아야 했으나, 그 결과로 학생들이 좋은 언어를 배울 수 있는 기회를 갖게 되었고, 게임 프로젝트도 훌륭히 수행한 것을 보게 되어 큰 보람을 느낀다.

미국의 주요 대학들이 수년전부터 프로그래밍 입문 언어로 파이썬을 채택하고 있는 점은 국내 컴퓨터 관련 및 게임 개발 관련 학과들이 눈여겨 살펴볼 필요가 있다.

게임 개발 측면에서 보면, 파이썬이 인터프리적인 특성 탓에 C/C++에 비해 메모리 사용량이라든가 수행속도 측면에서 단점을 갖고 있는 것은 분명하나, 해결할 수 있는 다양한 방법들이 알려져 있기 때문에 게임 개발에 충분히 파이썬이 활용될 수 있을 것으로 기대된다. 특히, 게임 개발의 교육에 있어서 파이썬의 가치와 효용성이 매우 높기 때문에, 앞으로 국내 게임 관련 학과에서 적극적으로 파이썬의 도입을 고려해 볼 것을 제언한다.

마지막으로, 지난 KGC 2014 컨퍼런스에서 필자가 강연했던 “한시간만에 끝내는 파이썬 게임 프로그래밍”의 발표자료 마지막 슬라이드의 내용을 인용하면서 본고를 맺고자 한다.

“배우기 쉽고, 이해하기 쉽고, 빨리 만들 수 있고, 써먹을 곳 많은 파이썬.... 왜 안배우시나요?”

참고문헌

- [1] “어린이코딩교육도우미, 스크래치 배워요”, 블로터닷넷 2014년 6월 29일.
<http://www.bloter.net/archives/197687>
- [2] Philip Guo, “Python is Now the Most Popular Introductory Teaching Language at Top U.S. Universities”, Communication of the ACM, Blogs, July 7, 2014.
<http://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-us-universities/fulltext>
- [3] A.B. Downey(2013), Think Python(pp.xii), O'REILLY

- [4] "Python at Google". <http://quintagroup.com/cms/python/google>
- [5] Stavros Korokithakis, "Tutorial - Learn Python in 10 minutes".
<http://www.stavros.io/tutorials/python/>

[저자정보]

이대현 (daehyun.lee@gmail.com) 한국산업기술대학교 게임공학부 부교수