

<http://dx.doi.org/10.17703/JCCT.2021.7.3.247>

JCCT 2021-8-29

온라인 학습 환경에서 발생하는 파이썬 프로그래밍 오류 사례 분석

A Case Study of Python Programming Error in an Online Learning Environment

정혜욱*

Hye-Wuk Jung*

요약 컴퓨터 프로그램 초보 학습자의 프로그래밍 실습과정에서 발생하는 프로그래밍 오류는 다양하다. 이때 학습자는 스스로 오류사항을 인지하기 어렵기 때문에 교수자의 피드백을 통해 프로그램 오류를 수정하게 된다. 그러나 최근 코로나19로 인해 온라인 환경에서 프로그래밍 기법을 학습하게 됨에 따라 오프라인 수업에 비해 교수자와의 상호작용에 한계가 있으므로 학습자 스스로 프로그래밍 오류를 해결하는 능력을 키울 필요가 있다. 이에 본 연구에서는 파이썬 언어를 이용한 온라인 프로그래밍 수업에서 발생된 학습자들의 오류 사례를 분석하고, 그 결과를 바탕으로 학습자의 프로그래밍 오류 수정 능력을 키워줄 수 있는 온라인 프로그래밍 교육 방안을 제시하였다.

주요어 : 온라인 학습, 파이썬 프로그래밍, 프로그래밍 오류, 소프트웨어 교육

Abstract There are various programming errors that occur in the course of programming practice for beginners in computer programming. At this time, since it is difficult for learners to recognize errors by themselves, they correct program errors through the instructor's feedback. However, as students learn programming techniques in an online learning environment due to the COVID-19 pandemic, there is a limit to interaction between the students and the instructor in comparison with offline classes, so it is necessary for learners to develop their own ability to solve programming errors by themselves. Therefore, in this study, error cases in online programming classes using the Python language are analyzed and an online programming education method that can improve learners' ability to correct programming errors is proposed based on the analysis results.

Key words : Online Learning, Python Programming, Programming Errors, Software Education

1. 서론

컴퓨터 프로그래밍에서 오류해결은 주어진 문제를 해결하고 원하는 결과를 정확히 얻기 위해 반드시 필요한 과정이다. 숙련된 프로그램 개발자는 코딩한 내용을

탐색하며 오류를 감지하고 수정하게 된다[1]. 그러나 프로그래밍을 처음 접하는 초보 학습자는 경험해 보지 않은 오류가 발생했을 때, 교수자의 피드백을 받아 해결하는 과정을 통해 오류수정 방법을 익힐 수 있다[2]. 이러한 오류 해결 과정은 프로그래밍 학습에서 매우 중요

*정혜원, 경기대학교 진성예교양대학 교양학부 조교수
접수일: 2021년 6월 30일, 수정완료일: 2021년 7월 23일
게재확정일: 2021년 7월 31일

Received: June 30, 2021 / Revised: July 23, 2021

Accepted: July 31, 2021

*Corresponding Author: wukj@kyonggi.ac.kr

Dept. of College of Liberal Arts and Interdisciplinary Studies,
kyonggi University, Korea

한 문제이기 때문에 학습자의 오류 유형을 분석하고 해결하는 방법과 관련 내용은 다양한 측면에서 연구되고 있다.

남재원과 유인환은 NXC 언어를 이용한 로봇 프로그래밍 학습에서 발생하는 학습자들의 오류유형을 분석하고, 이를 기반으로 프로그램 오류 해결에 참고할 수 있는 학습 지원 도구를 개발하여 수업에 적용시킨 후 학습자들의 프로그래밍 오류 발생이 감소됨을 확인하였다[3].

최현중은 C 프로그래밍 학습의 알고리즘 작성 과정에서 제어구조 설계에서 발생하는 오류를 바탕으로 오류유형을 제시하여 학습자들이 알고리즘 학습에 참고할 수 있는 방법을 제안하였다[4].

김지선과 김영식은 중·고등학생들을 대상으로 한 온라인 C 프로그래밍 학습과정에서 오류 데이터를 수집 및 분석하여 학습자의 성취수준과 오류유형의 상관관계를 기반으로 프로그래밍 학습 방향을 제시하였다[5].

문외식은 예비 교사들의 파이썬과 스크래치 소프트웨어 교육 과정에서 발생하는 오류 데이터를 수집하고 유형별로 분석하여 소프트웨어교육 능력을 향상시킬 수 있는 방법을 도출하였다[6].

강의선은 대학의 컴퓨터 비전공자들이 소프트웨어를 학습하는 과정에서 학습자의 특성 파악을 위해 파이썬 언어의 오류 메시지 해결 정도 등과 같은 어려운 부분을 분석하고 학습에 도움이 될 수 있는 구조적 소프트웨어 교육 모형을 제안하였다[7].

이와 같이 컴퓨터 프로그래밍 학습과정에서 발생하는 오류를 분석하고 해결하는 방법이 다양한 방향으로 연구되고 있지만, 최근 코로나19로 인해 온라인에서 컴퓨터 프로그래밍을 학습하는 상황에 대한 고려가 요구된다. 예를 들어, 학습자가 온라인에서 프로그래밍 실습을 진행하면서 오류가 발생했을 때, 그 원인을 신속하게 찾아 필요한 해결방법을 문의하기에는 오프라인 학습 환경에 비해 교수자와의 상호작용에 있어 지연시간, 의사소통 등에 한계점이 발생하게 된다. 또한 파이썬 언어에서 발생하는 오류의 종류와 이에 해당하는 오류 메시지는 매우 다양하기 때문에 학습자들이 스스로 오류를 해결하는데 어려움이 많이 생긴다.

따라서 온라인 환경의 학습자들이 프로그래밍 과정에서 다양한 유형의 오류를 경험하며 오류유형을 익히

고 해결하는 능력을 키울 필요가 있다[8].

본 연구에서는 파이썬 언어를 이용한 온라인 프로그래밍 수업에서 프로그램 초보 학습자들의 오류발생 유형에 대한 사례를 분석하고, 그 결과를 바탕으로 학습자의 프로그래밍 오류 수정 능력을 키워줄 수 있는 온라인 프로그래밍 교육 방안을 제시하고자 한다.

II. 연구방법 및 오류 유형 분류

1. 연구대상 및 방법

본 연구는 코로나19로 인해 온라인 수업이 진행된 2020년 1학기, 2020년 2학기, 2021년 1학기에 파이썬 프로그래밍을 학습한 경기도의 K대학 교양 필수 과목 수강자(인문, 사회과학, 예체능계열) 536명을 대상으로 하였다. 오류 데이터 수집은 실습과정에서 발생한 오류 항목을 조사하였고, 교과목 정보 및 내용은 표 1과 같다.

표 1. 교과목 정보 및 내용

Table 1. Subject Education Information and Content

교과목 정보	내 용
학습 언어	파이썬(Python)
학습 난이도	기초과정
학습 기간	15주
학습 단원	입출력, 변수, 연산자, 리스트, 조건문, 반복문, 함수, 모듈
실습 환경	IDLE 통합 개발 환경
수업 방법	이론 설명, 예제 풀이(소스코드 작성 시연), 응용문제 풀기(실습과제)

2. 오류 유형 분류

파이썬 프로그램에서 발생하는 논리적 오류(Logical Errors)는 프로그램이 잘못된 작업이나 순서로 인해 충돌하여 잘못 작동됨으로써 올바른 결과 값을 출력한다. 구문 오류(Syntax Errors)는 언어의 올바른 순서를 따르지 않았을 때 발생되며, 프로그램을 실행하기 전에 수정해야 정상적으로 프로그램이 동작된다. 구문 오류의 예로는 줄 끝의 세미콜론이나 괄호가 누락되거나 맞춤법이 잘못 입력된 경우가 있다. 또한 구문 오류가 없을 경우에도 프로그램이 인터프리터로 번역되어 실행하는 중에 발생하는 런타임 오류(Runtime Errors)가 있다[9, 10].

표 2는 본 연구에서 적용한 파이썬 학습 단원 내 실습과정에서 발생된 오류들을 기준으로 분류한 오류 유형이다. 오류 유형은 파이썬 소스코드 실행과정에서 출력되는 오류 메시지와 파이썬 홈페이지 문서의 오류 내용을 참고하여 분류하였고, 각 오류 유형을 식별하기 위해 유형 코드 항목을 생성하였다[11].

표 2. 오류 유형 분류 및 코드
 Table 2. Error Type Classification and Codes

오류 유형	유형코드	내 용
LogicalError	L_Err	프로그램의 순서와 작동 오류
SyntaxError	S_Err	프로그램의 문법 오류
NameError	N_Err	이름을 찾을 수 없는 오류
EOLError	EL_Err	문자열 표기 오류
EOFError	EF_Err	파일의 끝을 만나는 오류
TypeError	T_Err	잘못된 타입 적용 오류
AttributeError	A_Err	속성 할당 오류
ValueError	V_Err	잘못된 값의 전달 오류
TurtleGraphicsError	TG_Err	터틀그래픽 모듈 관련 오류
IndexError	Id_Err	인덱스 범위 설정 오류
IndentationError	Idt_Err	들여쓰기 블록 맞춤 오류

III. 오류 사례 분석

파이썬 단원별 학습과정에서 발생한 오류 사례는 표 3과 같고, 각 오류 유형은 표 2의 유형 코드로 나타냈다. 터틀그래픽의 경우 모듈에 속하지만, 수업에서 터틀 그래픽 예제를 다양하게 다루었으므로 모듈과 분리하여 오류내용을 파악하였다.

표 3. 파이썬 프로그램의 단원별 오류 사례
 Table 3. Error Cases by Chapter in Python Programs

파이썬 단원	오 류	
	유형 코드	내 용
출력	L_Err	대입 연산자(=) 표기
	S_Err	괄호() 누락
		섭표(.) 누락
	N_Err	따옴표(“) 누락
		변수명 불일치 print 함수명 오타
	A_Err	마침표(.) 사용
	EF_Err	괄호 오류(짝이 안 맞춤)
T_Err	자료형 변환 오류	
입력	S_Err	괄호 오류(짝 안 맞춤)
		따옴표(“) 누락
	N_Err	변수 사용 누락
		함수명 오타
V_Err	함수명 누락	
T_Err	자료형 변환 오류	

변수	S_Err	변수명 오류	
	N_Err	대입연산자 오류	
리스트	N_Err	변수명 오타	
	S_Err	따옴표(“) 누락	
	Id_Err	인덱스 괄호[] 표기 오류	
문자열 수식	Id_Err	인덱스 범위 설정 오류	
	L_Err	잘못된 수식오류	
	S_Err	잘못된 문장 부호	
조건문	EL_Err	문자열 따옴표 오류(짝 안 맞춤)	
	T_Err	문자열 슬라이스 추출 범위설정 오류	
	S_Err	콜론(:) 누락 및 세미콜론(;)으로 표기 elif 예약어 표기 오류 else문 조건 포함으로 인한 오류	
반복문	Idt_Err	들여쓰기 블록 맞춤 오류	
	S_Err	콜론(:) 누락 및 세미콜론(;)으로 표기 for문에서 in 누락 변수 설정 오류	
	N_Err	반복문 변수 초기화 누락 while문의 무한루프 표기 오류	
함수	Idt_Err	들여쓰기 블록 맞춤 오류	
		L_Err	range 함수 범위설정 오류 함수의 호출부분 누락 함수의 return문 누락 및 오류
		S_Err	사용자정의 함수 괄호() 누락 콜론(:) 누락 및 세미콜론(;)으로 표기 잘못된 사용자정의 함수명 매개변수 구분을 위한 섭표(.)오류 사용자 정의함수 선언 시 def 누락
	N_Err	내장 함수명 오타 함수 호출 시 함수명 오류	
		T_Err	range 함수의 범위설정 누락 인수 누락 및 매개변수 매치 오류 잘못된 인수 값 입력
	A_Err	함수 호출 시 마침표(.) 누락 함수 호출 시 섭표(.)로 표기 함수 호출 시 함수명 오타	
		Idt_Err	함수에서 들여쓰기 블록 맞춤 오류
		모듈	N_Err
	T_Err		잘못된 인수 값 입력
	A_Err		함수명 오타
터틀 그래픽	N_Err	함수 호출 시 마침표(.) 누락 함수 호출 시 섭표(.)로 표기 색상 함수의 색상명 따옴표(“) 오류	
		T_Err	goto 함수의 좌표 값 구분 섭표(.) 누락 goto 함수의 좌표 값 마침표(.)로 표기
			A_Err
	TG_Err	색상 함수의 색상명 오타 모양 함수의 모양명 오타	
		함수 사용 시 잘못된 인수 값 입력	

• 출력

출력문에서 발생하는 L_Err은 “print(=)”과 같이 대입 연산자(=)를 표기했을 때 나타난다. print() 함수의

괄호가 누락되거나 괄호 안에서 변수, 문자열 등을 구분하는 쉼표(,)가 누락된 경우에는 문법에 맞게 작성되지 않았기 때문에 S_Err이 발생한다. N_Err는 print() 함수 내 문자열 표기 시 따옴표(“) 누락, 선언된 변수명과 일치하지 않는 변수명 사용, 함수명에 오타가 있을 때 나타나는 오류이다. print() 함수 내 변수 등 구분시 쉼표 대신 마침표(.) 사용하면 A_Err, print 함수 괄호의 짝을 맞추지 않았을 때에는 EF_Err, print() 함수 내 자료형 변환을 잘못 적용하면 T_Err이 발생한다.

- 입력

input() 함수를 이용한 입력문에서는 괄호의 짝을 맞추지 않거나 함수 내 문자열 표기 시 따옴표(“)가 누락되었을 때 S_Err이 발생한다. N_Err은 입력 값을 대입하는 변수사용이 누락되거나 input 함수명에 오타가 있을 때 나타난다. input 함수명이 누락된 경우의 오류는 V_Err, 입력 값의 자료형 변환 함수를 잘못 적용했을 때 나타나는 오류는 T_Err이다.

- 변수

변수에서는 변수명 규칙에 맞지 않게 변수명을 선언(공백 등을 포함)하거나 대입연산자를 잘못 사용(==대신 = 표기)하면 S_Err 오류가 발생한다. 선언된 변수를 수식이나 출력문에서 사용할 때 오타가 있으면 N_Err이 나타난다.

- 리스트

리스트 데이터의 문자열 표기 시 따옴표(“)가 누락된 경우에는 N_Err, 인덱스 괄호[] 표기를 잘못했을 때는 S_Err, 리스트의 인덱스 범위 설정에 오류가 있으면 Id_Err이 발생한다.

- 문자열 및 수식

문자열 및 수식 부분에서는 잘못된 수식(연산자 우선순위, 연산자 표기, 조건식) 사용으로 정상적인 결과 값이 나오지 않는 L_Err과 잘못된 문장 부호 사용으로 인한 S_Err이 있다. 문자열 따옴표 오류(짝 안 맞추면) EL_Err이 문자열 슬라이스 추출 범위설정 오류 T_Err이 발생한다.

- 조건문

조건문에서 S_Err은 if문 끝에 콜론(:) 누락 및 세미콜론(;)으로 잘못 표기, elif 예약어 잘못 표기, else문에 조건을 포함한 경우에 나타난다. 조건 문장의 들여쓰기와 블록 맞추미 잘못되면 Idt_Err이 발생한다.

- 반복문

반복문에서는 for 또는 while문 끝에 콜론(:) 누락 및 세미콜론(;)으로 표기, for문에서 in 누락, for문에서 반복변수 설정 오류(변수 i를 숫자 1로 표기)로 인해 S_Err이 발생된다. N_Err은 변수의 초기화 누락되거나 while문의 무한루프 표기를 잘못한 경우에 나타난다. 반복 문장의 들여쓰기와 블록 맞추미 잘못되면 Idt_Err이 발생한다.

- 함수

함수에서 발생하는 L_Err은 반복문에서 사용하는 range 함수의 범위를 잘못 설정하거나, 사용자 정의 함수의 호출부분 누락, return문 누락 및 잘못 설정으로 인해 나타난다. S_Err은 사용자 정의 함수 괄호() 누락, 함수명 선언 끝부분의 콜론(:) 누락 및 세미콜론(;)으로 표기, 잘못된 함수명, def가 누락된 경우와 매개변수가 있는 함수에서 여러 개의 매개변수 구분을 위한 쉼표(,)를 사용하지 않아 발생한다. 내장 함수명에 오타가 있거나 사용자 정의 함수 호출 시 함수명 오류가 있으면 N_Err이 발생한다. T_Err은 range 함수의 범위설정 누락, 사용자 정의 함수 호출 시 인수(Argument) 값 누락 및 매개변수(Parameter)와의 매치(Match) 오류, 내장 함수 사용 시 잘못된 인수 값 입력에 의해 발생된다. 함수 호출 시 마침표(.) 누락 및 쉼표(,)로 잘못표기, 함수명에 오타가 있으면 A_Err이 나타나고, 함수에서 들여쓰기와 블록 맞추미 잘못되면 Idt_Err이 발생한다.

- 모듈

N_Err은 모듈 사용 시 import문 누락, 호출 변수가 누락 되었거나 잘못 표기 했을 때 나타난다. 모듈의 함수 사용 시 잘못된 인수 값이 입력되면 T_Err, 모듈의 함수 호출 시 함수명에 오타가 있으면 A_Err이 발생된다.

- 터틀그래픽

N_Err은 터틀그래픽 모듈의 함수 호출 시 마침표(.) 누락 및 쉼표(,)로 잘못표기, 색상 함수의 색상명 따옴표(“)가 누락되었을 때 나타나는 오류이다. 터틀그래픽 모듈의 goto 함수의 좌표 값 구분 쉼표(,) 누락 및 마침표(.)로 잘못표기 하면 T_Err이 발생하고, 터틀그래픽 모듈의 함수 괄호()가 누락되거나 함수 호출 시 함수명에 오타가 있으면 A_Err이 나타난다. TG_Err은 터틀그래픽 모듈의 색상 함수의 색상명 오타, 모양 함수의 모양명 오타, 함수 사용 시 잘못된 인수 값을 입력했을 때 발생한다.

이 밖에도 print, input 등 함수를 사용할 때 괄호를 마지막에 하나 더 쓴 경우에는 SyntaxError에 해당되며 'unmatched ')와 같은 오류 메시지가 발생된다. 프로그램을 실행할 때 값을 잘못 입력하여 ValueError가 발생되기도 하고 소스코드의 첫 칸에 공백이 있거나 변수명, 함수명 등의 띄어쓰기가 잘못된 경우에도 SyntaxError가 나타난다.

파이썬 프로그램 학습의 초반에 다루는 출력과 입력 부분에서는 괄호나 따옴표 등과 같이 기본적인 표기방법에 대한 오류가 많았다. 중반부에서는 변수선언 규칙에 대한 오류와 리스트 사용방법에 대한 오류 등이 있었고, 조건문과 반복문에서도 사용규칙에 맞지 않게 작성하는 경우가 많았다. 후반부에 다루는 함수에서는 함수 선언이나 호출 부분에서 발생하는 오류가 많았고, 모듈과 터틀그래픽에서도 잘못된 함수사용으로 인한 오류가 많이 나타났다. 또한 조건문, 반복문, 함수를 복합적으로 사용하는 경우에는 들여쓰기를 정확하게 하지 않아 블록 맞춤 오류가 발생하였다.

이와 같이 프로그래밍 초보 학습자의 실습과정에서는 파이썬 프로그램의 규정된 사용법과 일치하지 않을 때 발생하는 오류들이 많았다.

오류 확인은 소스코드를 실행할 때 나타나는 윈도우 알림창과 IDLE(Integrated Development and Learning Environment) 셸창의 붉은색 오류 메시지로 할 수 있다. 예를 들어, 셸창에서는 "NameError: name 'a' is not defined"와 같이 오류 유형인 NameError와 이름이 정의되지 않은 것에 대한 메시지가 출력된다. 논리적 오류의 경우에는 셸창에 결과 값이 나오지 않거나 "None"이라는 값이 출력된다. 또한 잘못된 결과 값이 산출되기도 한다.

IV. 온라인 프로그래밍 교육 방안

앞 절에서 다룬 오류 사례를 기반으로 실습과정에서 발생한 오류 중 빈번하게 나타나는 오류항목에 대한 설문조사를 한 결과는 그림 1과 같다.

전체 536명 중 120명(22.4%)이 8번(예약어, 함수명 등의 오타)을 선택하여 가장 많은 응답률을 보였고, 3번(조건문, 반복문, 함수 마지막의 콜론 :)/102명(19.0%), 4번(콤마(,)와 마침표(.) 표기)/93명(17.4%), 1번(괄호() 누락, 짝 안 맞춤)/92명(17.1%) 순으로 나타났다.

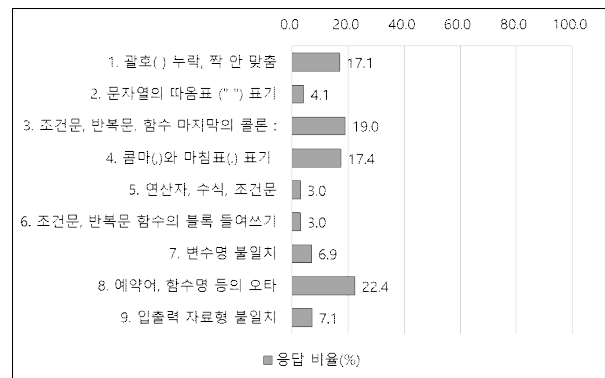


그림 1. 프로그램에서 자주 발생하는 오류항목
 Figure 1. Error Items Occurring Frequently in the Program

오류항목에 대한 일부 학습자들의 서술형 의견은 다음과 같다.

- 변수명이나 함수명의 오타로 인한 오류로 실행되지 않는 경우가 많다.
- 명령어를 입력하는데 영어를 잘못 입력하여 오류가 발생한다.
- 오타는 계속 반복적으로 발생한다. 예를 들어 급하게 타자를 치다보니 print를 pirnt와 같이 쓸 때가 종종 있다.
- 사용자 정의 함수 선언문 끝에 콜론(:)을 써야만 하는 경우가 있는데, 함수가 복잡해지면 다른 부분을 신경 쓰다가 잊고 쓰지 않아서 오류가 발생한다.
- 함수가 길어지고 복잡해질수록 더 꼼꼼하게 확인해야 할 것 같다.
- 함수를 실행시켰을 때 정상적으로 작동하지 않는다면, 어느 부분에서 오류가 발생한 것인지를 찾아낼 수 있도록 파이썬에 익숙해지고 발전할 수 있도록 노력해야 할 것 같다.

이러한 설문조사 결과로부터 파이썬 프로그램의 기본적인 사용 규칙에 대한 오류 항목이 실습과정에서 많이 발생되는 것을 알 수 있다. 파이썬 단원의 후반부도 갈수록, 길어지는 소스코드 작성 시에도 예약어, 함수명 등의 오타나 콜론(:), 콤마(,), 마침표(.)와 같은 표기법에 주의를 해야 한다. 또한 함수 사용 시 괄호()가 누락되지 않고 짝을 잘 맞춰 주지 않는 문제들이 나타난다.

이와 같은 오류 항목들은 프로그램 초보 학습자뿐만 아니라 일반적으로 프로그래밍 과정에서 반복적으로 많이 발생하는 오류이기도 하다. 그러나 초보 학습자들은

프로그래밍 작업에 익숙하지 않기 때문에 스스로 오류를 인지하고 수정하는데 어려움을 느끼게 된다. 특히 온라인 학습 환경에서는 교수자의 피드백을 신속하게 받는 데는 한계가 있다. 따라서 프로그래밍 오류가 발생되지 않게 프로그램 언어의 규칙을 정확하게 익히는 것도 중요하지만, 발생한 오류를 스스로 인지하고 해결하는 방법에 대한 교육이 필요하다. 그러므로 다음과 같이 온라인 프로그래밍 교육 방안을 제시하고자 한다.

첫째, 프로그래밍 단원별 문법규칙에 대한 설명과 함께 각 단원에서 발생할 수 있는 오류와 오류 메시지를 해석하는 방법에 대한 설명이 필요하다. 이러한 학습내용은 학습자들이 오류 유형을 익히고 해결하는데 도움이 될 것이다.

둘째, 소스코드의 오류를 빠르게 찾지 못할 경우에 단계별로 오류 사항을 확인해 나가며 오류를 해결하는 방법을 익히는 것도 중요하다. 따라서 실습 예제 설명과정에서 오류 발생 상황을 교수자가 직접 시연하면서 오류를 찾아가며 해결하는 과정을 보여주게 되면 학습자 혼자 실습하는 환경에서도 오류 해결을 잘 할 수 있을 것이다.

셋째, 같은 내용의 오류에 서로 다른 오류 메시지가 발생되었을 때, 예를 들어 괄호()가 누락되면 사용자 정의 함수에서는 SyntaxError, 터틀그래픽에서는 AttributeError 오류 메시지가 나타난다. 이러한 경우 오류 유형을 파악할 수 있는 자료를 제공하거나 인터넷 등을 통해 참고할 수 있는 방법을 알려주면 학습자들이 스스로 오류를 이해하고 해결하는 작업에 도움이 될 것이다.

V. 결 론

최근 코로나 19 팬데믹으로 인해 프로그램 초보 학습자들의 프로그래밍 수업 환경이 오프라인에서 온라인으로 바뀌어 학습자가 작성한 프로그램에서 발생하는 오류사항을 교수자의 피드백을 통해 해결하는데 제약이 있는 상황이 되었다. 이러한 상황에서는 학습자 스스로 프로그래밍 오류를 해결할 수 있는 능력을 키우는 것이 매우 중요하다.

본 연구에서는 학습자들의 온라인 파이썬 프로그래밍 수업에서 발생한 오류 사례를 분석하고 그 결과를 기반으로 학습자의 프로그래밍 오류 수정 능력을 키워 줄 수 있는 다음과 같은 온라인 프로그래밍 교육 방안을 제시하였다.

첫째, 프로그래밍 단원별 문법규칙에 대한 설명과 함께 각 단원에서 발생할 수 있는 오류와 오류 메시지를 해석하는 방법에 대한 설명이 필요하다. 둘째, 소스코드의 오류를 빠르게 찾지 못할 경우에 단계별로 오류 사항을 확인해 나가며 오류를 해결하는 방법을 익히도록 학습하는 것이 필요하다. 셋째, 같은 내용의 오류에 서로 다른 오류 메시지가 발생되었을 경우, 오류 유형을 파악할 수 있는 자료 또는 인터넷 등을 통해 참고할 수 있는 방법을 제공하는 것이 필요하다.

이와 같은 학습 과정을 통해 학습자가 프로그램 작성에서 발생하는 오류들을 스스로 해결할 수 있는 능력이 향상될 것으로 기대한다. 향후에는 이러한 학습 방법이 학습자들의 오류해결 능력에 미치는 영향을 추가적으로 분석해야 할 필요성이 있다.

References

- [1] A. J. Ko, B. A. Myers, "Development and evaluation of a model of programming errors", IEEE Symposium on Human Centric Computing Languages and Environments, 2003. DOI: <https://doi.org/10.1109/HCC.2003.1260196>
- [2] H. W. Jung, "A Study on Teacher-learner Feedback Method for Effective Software Project Execution of Non-Computer Major Students," The Journal of the Convergence on Culture Technology (JCCT), Vol. 5, No. 1, pp. 211-217, 2019. DOI: <https://dx.doi.org/10.17703/JCCT.2019.5.1.211>
- [3] J. W. Nam and I. H. Yoo, "Development of NXC Robot Programming Supporting System Based on Types of Programming Error," Journal of The Korean Association of Information Education, Vol. 15, No. 3, pp. 375 - 385, 2011.
- [4] H. Choe, "Analysis on Types of Errors in Learning about Control Structures of Programming using Flowchart," The Journal of Korean Association of Computer Education, Vol. 19, No. 1, pp. 101 - 109, 2016.
- [5] J. Kim and Y. Kim, "The Analysis of Relationship between Academic Achievement Level of Concept Learning and Error Type in Online Programming Course," The Journal of Korean Association of Computer Education, Vol. 17, No. 5, pp. 43 -51, 2014.
- [6] W. S. Moon, "Analysis of error data generated by prospective teachers in programming learning,"

- Journal of The Korean Association of Information Education, Vol. 22, No. 2, pp. 205-212, 2018. DOI: <https://dx.doi.org/10.14352/jkaie.2018.22.2.205>
- [7] E. Kang, “Structural Software Education Model for Non-majors - Focused on Python,” Journal of Digital Contents Society, Vol. 20, No. 12, pp. 2423-2432, 2019.
- [8] H. W. Jung, “A Study on Coding Education for Non-Computer Majors Using Programming Error List ,” International Journal of Advanced Culture Technology, Vol. 9, No. 1, pp. 203-209, 2021. DOI: <https://doi.org/10.17703/IJACT.2021.9.1.203>
- [9] G. Samara, “A Practical Approach for Detecting Logical Error in Object Oriented Environment ,” World of Computer Science and Information Technology Journal (WCSIT), Vol. 7, No. 2, pp. 10-19, 2017.
- [10]<https://docs.python.org/3/tutorial/errors.html>
- [11]<https://docs.python.org/3/library/exceptions.html>

※ 이 논문은 대한민국 교육부와 한국연구재단
의 2017년도 이공학 개인기초연구지원사업
(No. 2015R1D1A1A01061064)과 2021년
도 인문사회분야 신진연구자지원사업(NR
F-2020S1A5A8042850)의 지원을 받아
수행된 연구임