(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0148255
(43) 공개일자 2024년10월11일

(51) 국제특허분류(Int. Cl.)
G09B 19/00 (2006.01) *G06Q 50/20* (2012.01)
G09B 5/02 (2006.01) *G09B 7/06* (2006.01)
(52) CPC특허분류
G09B 19/0053 (2013.01)
G06Q 50/20 (2013.01)
(21) 출원번호 10-2023-0196281
(22) 출원일자 2023년12월29일
심사청구일자 2023년12월29일
(30) 우선권주장
1020230043544 2023년04월03일 대한민국(KR)

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
김자미
서울특별시 용산구 한남대로46길 18, 501호 (한남동, 주노빌)
김용천
서울특별시 용산구 이촌로 201, 202동 606호 (이촌동, 한가람아파트)
(74) 대리인
김등용

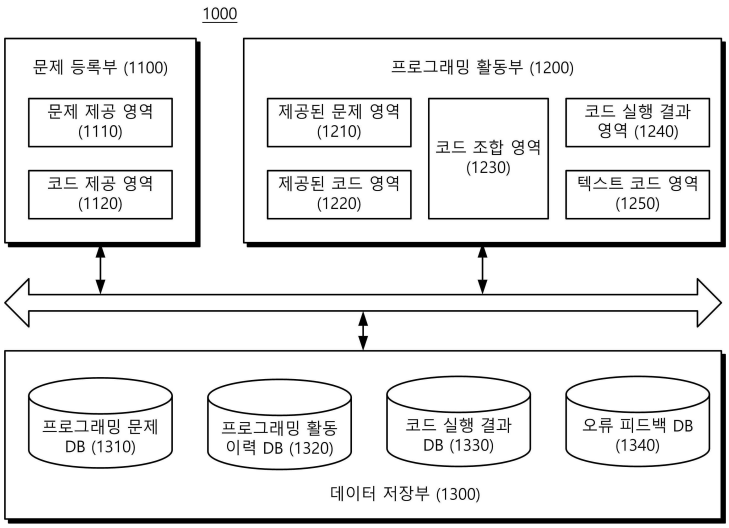
전체 청구항 수 : 총 4 항

(54) 발명의 명칭 **블록기반 파이썬 프로그래밍 학습 시스템**

(57) 요 약

본 발명의 실시 예에 따른 파이썬 프로그래밍 학습 시스템은, 파이썬으로 만들어야 하는 프로그램의 제목과 내용을 작성하기 위한 문제 등록부; 문제 등록자가 제공한 코드를 드래그 앤 드롭으로 조합하고, 조합된 블록 형태의 코드를 텍스트 코드로 보여주기 위한 프로그래밍 활동부; 및 파이썬으로 프로그램을 만들 때 사용되는 코드를 저장하기 위한 데이터 저장부를 포함할 수 있다. 상기 프로그래밍 활동부는 상기 블록 형태의 코드가 이동, 추가, 수정 및 삭제될 때마다 텍스트 코드 영역의 텍스트 코드도 동시에 변경될 수 있다.

대 표 도



(52) CPC특허분류
G09B 5/02 (2013.01)
G09B 7/06 (2013.01)

명 세 서

청구범위

청구항 1

파이썬 프로그래밍 학습 시스템에 있어서,

파이썬으로 만들어야 하는 프로그램의 제목과 내용을 작성하기 위한 문제 등록부;

문제 등록자가 제공한 코드를 드래그 앤 드롭으로 조합하고, 조합된 블록 형태의 코드를 텍스트 코드로 보여주기 위한 프로그래밍 활동부; 및

파이썬으로 프로그램을 만들 때 사용되는 코드를 저장하기 위한 데이터 저장부를 포함하되,

상기 프로그래밍 활동부는 상기 블록 형태의 코드가 이동, 추가, 수정 및 삭제될 때마다 텍스트 코드 영역의 텍스트 코드도 동시에 변경되는 파이썬 프로그래밍 학습 시스템.

청구항 2

제1 항에 있어서,

상기 문제 등록부는 제공된 명령어의 무작위 순서 배치 설정부, 제공된 명령어 수정 설정부, 그리고 제공된 명령어 외의 명령어를 추가하기 위한 명령어 추가 설정부를 포함하는 파이썬 프로그래밍 학습 시스템.

청구항 3

제2 항에 있어서,

상기 제공된 명령어 수정 설정부는 제공된 명령어의 수정, 제공된 명령어의 수정 완료, 제공된 명령어의 삭제를 하도록 제어하는 파이썬 프로그래밍 학습 시스템.

청구항 4

제2 항에 있어서,

상기 제공된 명령어 추가 설정부는 추가할 명령어의 코드와 설명글을 입력하도록 제어하는 파이썬 프로그래밍 학습 시스템.

발명의 설명

기 술 분 야

[0001] 본 발명은 프로그래밍 학습 시스템에 관한 것으로, 더욱 상세하게는 블록기반 파이썬 프로그래밍 학습 시스템에 관한 것이다.

배 경 기 술

[0002] 1990년대에는 입문자에게 C, Python, C++ 등 실제로 산업 현장에서 사용되는 텍스트 프로그래밍 언어를 가르쳤다. 텍스트 프로그래밍 언어는 입문자가 명령어를 키보드로 입력해야 하기 때문에 오타로 인한 오류가 발생했다. 텍스트 프로그래밍 언어는 사용할 수 있는 명령어를 시각적으로 제공하지 않기 때문에, 입문자는 어떤 명령어가 있는지 알기 어렵다.

[0003] 2006년에 소개된 스크래치(Scratch)는 사용 가능한 명령어를 시각적으로 제공해준다. 입문자는 명령어를 키보드로 입력하지 않고, 마우스로 조합하여 프로그래밍 활동을 할 수 있다. 스크래치는 입문자가 프로그램을 만들 수 있도록 도움을 준다. 그러나 스크래치는 자체적으로 개발한 명령어를 스크래치 사이트 내에서만 사용해야 한다. 입문자가 스크래치를 잘 사용한다고 해서 C, Python, C++ 등 산업 현장에서 사용되는 프로그래밍 언어를 사용할 수 있게 되는 것은 아니다.

[0004] 이러한 한계를 해결하기 위해 도 1과 같은 Pencil Code는 입문자가 블록으로 조합한 명령어를 텍스트 명령어로

변환해주는 기능을 제공한다. 그러나, 입문자가 텍스트 명령어를 보는 것만으로 프로그래밍 언어를 학습했다고 보기 어렵다. 프로그래밍 언어를 학습하기 위해서는 입문자가 명령어를 직접 입력해보고 수정, 삭제, 추가해보는 활동을 통해 익히는 과정이 필요하다.

발명의 내용

해결하려는 과제

[0005] 본 발명은 상술한 과제를 해결하기 위한 것으로서, 본 발명의 목적은 입문자의 프로그래밍 능숙도를 향상시키는데 기여할 수 있는 프로그래밍 환경을 제공하는 블록 기반의 파이썬 프로그래밍 학습 시스템을 제공하는 데 있다.

과제의 해결 수단

[0006] 본 발명의 실시 예에 따른 파이썬 프로그래밍 학습 시스템은, 파이썬으로 만들어야 하는 프로그램의 제목과 내용을 작성하기 위한 문제 등록부; 문제 등록자가 제공한 코드를 드래그 앤 드롭으로 조합하고, 조합된 블록 형태의 코드를 텍스트 코드로 보여주기 위한 프로그래밍 활동부; 및 파이썬으로 프로그램을 만들 때 사용되는 코드를 저장하기 위한 데이터 저장부를 포함할 수 있다. 상기 프로그래밍 활동부는 상기 블록 형태의 코드가 이동, 추가, 수정 및 삭제될 때마다 텍스트 코드 영역의 텍스트 코드도 동시에 변경될 수 있다.

[0007] 실시 예로서, 상기 문제 등록부는 제공된 명령어의 무작위 순서 배치 설정부, 제공된 명령어 수정 설정부, 그리고 제공된 명령어 외의 명령어를 추가하기 위한 명령어 추가 설정부를 포함할 수 있다.

[0008] 실시 예로서, 상기 제공된 명령어 수정 설정부는 제공된 명령어의 수정, 제공된 명령어의 수정 완료, 제공된 명령어의 삭제를 하도록 제어할 수 있다. 상기 제공된 명령어 추가 설정부는 추가할 명령어의 코드와 설명글을 입력하도록 제어할 수 있다.

[0009] 실시 예로서, 상기 프로그래밍 활동부는 제공된 명령어의 한 줄씩 실행부, 제공된 명령어의 전체 실행부, 제공된 명령어와 정답으로 등록된 코드와 비교하기 위한 코드 비교부, 그리고 제공된 명령어를 초기화 하기 위한 명령어 초기화부를 포함할 수 있다.

[0010] 실시 예로서, 상기 프로그래밍 활동부는 명령어를 이동하면 조합된 코드가 텍스트 형태로 표현되는 조합된 코드의 텍스트 형태 목록부를 포함할 수 있다. 상기 프로그래밍 활동부는 오류의 위치가 시각적으로 표시되는 오류 위치 표시부를 포함할 수 있다.

발명의 효과

[0011] 본 발명에 의하면, 입문자는 파이썬 프로그래밍의 명령어를 읽고, 쓰고, 수정하면서 쉽게 익히고 배울 수 있다.

도면의 간단한 설명

[0012] 도 1은 종래의 프로그래밍 학습 방법을 예시적으로 설명하기 위한 도면이다.

도 2는 Parsons Problems의 예시 문제를 나타낸 도면이다.

도 3 및 도 4는 외국어를 배우는 모델에 기초하여 프로그래밍 언어를 학습하는 과정을 보여주는 알고리즘이다.

도 5는 본 발명의 실시 예에 따른 파이썬 프로그래밍 학습 시스템을 예시적으로 보여주는 블록도이다.

도 6은 도 5에 도시된 문제 등록부를 예시적으로 보여주는 화면이다.

도 7은 도 5에 도시된 프로그래밍 활동부를 예시적으로 보여주는 화면이다.

도 8 내지 도 12는 도 7에 도시된 코드 조합 영역을 예시적으로 보여주는 화면이다.

도 13은 프로그래밍 요소의 문제 풀이 순서를 나타낸 도면이다.

도 14는 문제 풀이 결과를 나타낸 도면이다.

발명을 실시하기 위한 구체적인 내용

[0013] 이하에서, 본 발명의 기술 분야에서 통상의 지식을 가진 자가 본 발명을 용이하게 실시할 수 있을 정도로, 본

발명의 실시 예들이 명확하고 상세하게 기재될 것이다.

[0014] 도 2는 Parsons Problems의 예시 문제를 나타낸 도면이다. 왼쪽에 제시된 6개의 명령어를 올바른 순서대로 오른쪽 공간으로 이동시켜야 하며, 들여쓰기도 고려해야 한다. “feedback” 버튼을 눌러 정답 여부를 확인할 수 있다.

[0015] Parsons Problems는 다음과 같은 장점이 있다. 첫째, Parsons Problems는 스크래치와 같이 명령어를 마우스로 조합하여 프로그래밍할 수 있다. 둘째, Parsons Problems는 스크래치와 달리, 파이썬 명령어를 사용하기 때문에 텍스트 프로그래밍 언어로 전이하는데 도움이 될 수 있다. 셋째, 오류가 발생한 부분을 색깔로 표시하여 입문자가 오류 발생 원인을 빠르고 정확하게 인식하게 해준다.

[0016] 그러나, Parsons Problems는 제공된 명령어의 순서를 조합하는 기능만 제공할 뿐, 입문자가 명령어를 추가, 수정, 삭제하는 기능을 제공하지 않는다. 입문자가 순서를 조합하는 과정만으로 프로그래밍 학습이 이루어졌다고 보기 힘들다. 프로그래밍 학습이 이루어지기 위해서는 사용자가 직접 코드를 입력하고 수정하고 삭제하고, 이 과정에서 발생하는 오류를 해결하는 과정을 통해 프로그래밍 학습이 이루어져야 한다.

[0017] 입문자의 프로그래밍 학습 효과를 위해서는 입문자 스스로 프로그래밍에 대한 능숙함의 단계를 향상시킬 수 있는 환경을 제공할 필요가 있다. 즉, 블록 기반 프로그래밍의 익숙함이 프로그래밍에 대한 능숙함으로 이어지도록 할 필요가 있다.

[0018] 프로그래밍 과정에서 명령어를 추가, 수정, 삭제하는 과정의 중요성은 외국어를 배우는 과정과 연결하여 설명할 수 있다. 프로그래밍 학습 방법을 외국어를 배우는 모델에 기초한 연구에서는 단어나 문장이 누락된 빈 칸이 포함된 질문을 제공하고 학생들이 공백을 채우는 활동(Gapped text)을 하거나 목록에서 제공한 단어를 사용하여 올바른 문장을 완성하는 Jumbled sentences 활동을 제안한다.

[0019] 입문자가 프로그래밍 개념을 이해하지 못한 경우 이해를 돕기 위한 낮은 수준의 학습 단계가 제공되어야 한다. 인지적 견습모델(Cognitive Apprenticeship)에서는 프로그램을 만드는 것을 많은 시간이 걸리는 기술이라고 보고 그 기술을 배울 수 있는 방법 중 하나로 전문가의 곁에서 배우는 것을 강조한다.

[0020] 실제로, 전문가로부터 모범 사례를 통해 학습한 결과 프로그래밍 활동을 포기하는 비율이 감소하였다. 프로그래밍 학습 초기에 따라하기를 통해 프로그래밍 명령어에 초점을 맞춘 후 점차 프로그래밍 심화 과정까지 이어지도록 한 연구도 진행되고 있다. 대학교 입문자를 대상으로 프로그래밍 예제를 참고하여 학습하는 것이 프로그래밍 학습에 긍정적이라는 연구도 진행되었다. 대학교 입문자가 선호하는 프로그래밍 학습 방법 중 하나가 반복된 연습이라는 연구 결과도 있다.

[0021] 도 3 및 도 4는 외국어를 배우는 모델에 기초하여 프로그래밍 언어를 학습하는 과정을 보여주는 알고리즘이다. 도 3은 Gapped text(공백 채우기)을 보여준다. 단어나 문장이 누락된 빈칸이 포함된 질문을 제공하고 학생들이 채우는 활동이다. 도 4는 Jumbled sentences(뒤죽박죽 문장)를 보여준다. 목록에서 선택한 단어나 문장을 사용하여 올바른 텍스트를 완성해야 하는 활동이다.

[0022] 도 3 및 도 4에 도시된 방식 외에도, 제시된 두 개의 목록 중 한 목록의 용어를 다른 목록의 용어 또는 정의와 일치시키는 Multiple matching(다중 일치) 활동이 있다. 또한, 정답 목록이 제공되고 학생은 정답을 하나 이상 선택하는 Multiple choice(객관식) 활동이 있다.

[0023] 이러한 유형은 입문자가 명령어를 입력하지 않고 순서를 배치하면서 명령어에 익숙해지도록 하는 장점이 있다. 전체 명령어를 작성하지 않고 핵심적인 부분만 입력하도록 함으로써 명령어 입력에 따른 인지적 부담과 오류 발생 횟수를 줄이는 장점이 있다.

[0024] 기존의 프로그래밍 환경은 입문자의 수준과 학습 과정에 대한 기능을 제공하고 있지 않다. 입문자를 위한 프로그래밍 환경에서는 외국어를 배우는 모델에 기초한 문제 풀이 방식을 적용하여 스스로 텍스트 기반 프로그래밍 언어를 사용할 수 있도록 할 필요가 있다.

[0025] 본 발명은 기존 연구 분석 및 전문가 의견 수렴을 통해 파이썬 입문자의 학습에 도움을 주는 프로그래밍 환경을 개발하기 위한 전략을 다음과 같이 구성하였다.

[0026] 첫째, 프로그래밍 환경은 학습과 평가가 동시에 이루어질 수 있어야 한다. 프로그래밍의 경우, 이론과 실습을 분리하기보다 두 가지를 동시에 경험하면서 실시간으로 피드백을 받을 때 학습의 효과를 높일 수 있다. 따라서 학습이 이루어지는 과정에서 평가를 통한 피드백이 동시에 진행될 필요가 있다.

[0027] 둘째, 프로그래밍 환경은 입문자 맞춤형이 가능해야 한다. 프로그래밍 학습에서 입문자의 수준과 맞지 않을 경우, 다음 단계로의 활동이 불가능할 수 있기 때문에 맞춤형 학습이 중요하다. 따라서 입문자의 수준을 고려하여 따라하기, 수정하기, 추가하기 등과 같이 단계를 구성하여 입문자가 자신의 수준에 적합한 단계를 선택할 수 있도록 구성할 필요가 있다.

[0028] 예를 들어, 외국어를 학습하는 과정과 비슷하게 입문자가 직접 명령어를 입력하고, 수정하고, 추가해보는 과정이 제공되어야 한다. 따라하기 수준에서 시작하여 Parsons Problems와 같이 명령어의 순서를 조합하는 과정을 통해 명령어에 익숙해지도록 할 필요가 있다. 그리고 사용자가 명령어를 직접 수정하는 과정과 명령어 한 줄을 추가하는 과정을 구성할 수 있는 기능이 필요하다.

[0029] 셋째, 프로그래밍 환경은 프로그래밍의 순서를 고려한 학습을 구성할 수 있어야 한다. 입문자가 프로그래밍 언어에 대해 체계적으로 학습하고, 프로그래밍에 대한 이해를 넓히기 위해서는 입력, 출력, 조건, 반복 등 프로그래밍 요소의 학습 순서를 단계별로 제공할 필요가 있다. 이 과정에서 입문자의 다양한 수준을 함께 고려하면, 뒤에 남겨진 입문자 없이 완전 학습이 가능할 것으로 판단된다.

[0030] 넷째, 프로그래밍 환경은 오류 발생 위치의 시각적 제공 및 모국어 기반 오류 메시지를 제공해야 한다. 입문자는 오류가 발생한 경우, 오류의 위치 및 오류 원인을 알기 어렵다. 특히, 영어를 모국어로 사용하지 않는 나라의 학생은 오류의 내용을 이해하는데 더 많은 어려움을 느낀다. 따라서, 오류 내용의 모국어 변환 및 Parsons Problems와 같이 오류가 발생한 위치를 색으로 표시해 줄 필요가 있다.

[0031] 도 5는 본 발명의 실시 예에 따른 파이썬 프로그래밍 학습 시스템을 예시적으로 보여주는 블록도이다. 도 5를 참조하면, 파이썬 프로그래밍 학습 시스템(1000)은 문제 등록부(1100), 프로그래밍 활동부(1200), 그리고 데이터 저장부(1300)를 포함한다.

[0032] 문제 등록부(1100)는 문제 제공 영역(1110)과 코드 제공 영역(1120)을 포함한다. 문제 등록자는 문제 제공 영역(1110)에 입문자가 파이썬으로 만들어야 하는 프로그램의 제목과 내용을 작성할 수 있다. 문제 등록자는 등록한 문제를 입문자에게 공개할 지 여부도 설정할 수 있다.

[0033] 코드 제공 영역(1120)은 문제 등록자가 입문자에게 제공하는 코드를 작성하는 부분이다. 실시 예에 따라 코드 제공 영역(1120)에는 아무것도 작성하지 않도록 설정할 수도 있다. 문제 등록자는 입문자에게 제공하는 코드를 등록한 순서대로 제공할지, 제공하는 코드의 순서를 무작위로 섞어서 제공할 지 여부 등을 설정할 수 있다. 문제 등록자가 입문자에게 제공한 코드를 입문자가 수정할 수 있도록 할 지 여부를 설정할 수 있다. 문제 등록자가 입문자에게 제공한 코드 외에, 입문자가 직접 코드를 추가할 수 있도록 할 지 여부를 설정할 수 있다.

[0034] 프로그래밍 활동부(1200)는 제공된 문제 영역(1210), 제공된 코드 영역(1220), 코드 조합 영역(1230), 코드 실행 결과 영역(1240), 그리고 텍스트 코드 영역(1250)을 포함한다.

[0035] 제공된 문제 영역(1210)은 문제 등록자가 작성한 문제의 내용 부분으로, 입문자가 파이썬으로 만들어야 하는 프로그램의 내용이다.

[0036] 제공된 코드 영역(1220)은 문제 등록자가 입문자에게 제공한 코드가 보이는 부분이다. 설정한 결과에 따라, 문제 등록자가 입문자에게 제공한 코드가 등록된 순서대로 입문자에게 제공되거나, 무작위로 섞여서 제공될 수 있다. 문제 등록자가 입문자에게 제공한 코드를 입문자가 수정할 수 있도록 설정한 경우, 입문자는 문제 등록자가 제공한 코드를 수정할 수 있다. 문제 등록자가 입문자에게 제공한 코드 외에, 입문자가 직접 코드를 추가할 수 있도록 설정한 경우, 입문자는 코드를 직접 추가할 수 있다.

[0037] 코드 조합 영역(1230)은 문제 등록자가 제공한 코드 또는 입문자가 직접 추가한 코드를 마우스로 드래그 앤 드롭하여 조합하는 부분이다. 코드의 위치를 위나 아래로 이동할 수 있으며, 들여쓰기와 같이 하나의 코드를 다른 코드의 하위로 이동할 수도 있다.

[0038] 코드 실행 결과 영역(1240)은 조합된 코드에 오류가 없는 경우 실행 결과가 출력되고 오류가 있는 경우 오류 메시지가 출력되는 부분이다.

[0039] 텍스트 코드 영역(1250)은 조합된 블록 형태의 코드를 텍스트 코드로 보여주는 부분이다. 코드 실행 결과 영역(1240)에서 블록 형태의 코드가 이동, 추가, 수정, 삭제될 때마다 텍스트 코드 영역(1250)의 텍스트 코드도 동시에 변경된다.

[0040] 데이터 저장부(1300)는 프로그래밍 문제 DB(1310), 프로그래밍 활동 이력 DB(1320), 코드 실행 결과 DB(1330),

그리고 오류 피드백 DB(1340)를 포함한다.

[0041] 프로그래밍 문제 DB(1310)는 문제 등록자가 작성한 문제의 내용과 입문자가 파이썬으로 프로그램을 만들 때 활용할 수 있는 코드를 저장하는 곳이다. 문제 등록자가 입문자에게 제공한 코드를 등록한 순서대로 제공할 지, 무작위 순서로 제공할지 여부, 문제 등록자가 입문자에게 제공한 코드를 입문자가 수정하거나 추가할 수 있도록 할 지 여부의 내용도 저장된다.

[0042] 프로그래밍 활동 이력 DB(1320)는 입문자가 코드 조합 영역(1230)에서 코드를 이동, 추가, 수정, 삭제한 내용을 저장하는 곳이다.

[0043] 코드 실행 결과 DB(1330)는 입문자가 코드 조합 영역(1230)에서 조합한 코드의 실행했을 때, 오류가 발생하지 않은 경우 코드의 실행 결과를 저장하는 곳이다.

[0044] 오류 피드백 DB(1340)는 입문자가 코드 조합 영역(1230)에서 조합한 코드를 실행했을 때, 오류가 발생한 경우 오류 메시지가 저장되는 곳이다.

[0045] 도 6은 도 5에 도시된 문제 등록부를 예시적으로 보여주는 화면이다. 도 6을 참조하면, 문제 등록부(1100)는 문제 제공 영역(1110)과 코드 제공 영역(1120)을 포함한다.

[0046] 문제 제공 영역(1110)은 문제 제목 입력부(1111), 문제 공개 여부 선택부(1112), 문제 대표 이미지 선택부(1113), 문제 내용 입력부(1114), 문제 힌트 입력부(1115), 제공된 명령어의 무작위 순서 배치부(1116), 제공된 명령어 수정부(1117), 그리고 제공된 명령어 외의 명령어를 추가하기 위한 명령어 추가부(1118)를 포함한다.

[0047] 문제 제목 입력부(1111)는 도 13의 문제의 제목(1262)과 같이 다른 사용자에게 보여지는 문제 제목을 입력하는 부분이다. 문제 공개 여부 선택부(1112)는 도 13의 문제 목록(1261)과 같이 다른 사용자에게 보여지게 할지 보여지게 하지 않을지 선택하는 부분이다. 문제 대표 이미지 선택부(1113)는 도 13의 문제의 대표 이미지(1263)와 같이 다른 사용자에게 보여지는 문제의 대표 이미지를 업로드 하는 부분이다.

[0048] 문제 내용 입력부(1114)는 도 7의 제공된 문제 영역(1210)에 표시되는 내용을 입력하는 부분이다. 문제 힌트 입력부(1115)는 도 7의 제공된 문제의 힌트(1211)에 표시되는 내용을 입력하는 부분이다. 제공된 명령어의 무작위 순서 배치부(1116)는 도 7의 제공된 코드 영역(1220)의 명령어의 순서를 무작위로 섞어서 제공할지, 처음 등록된 그대로 제공할지를 선택하는 부분이다.

[0049] 제공된 명령어의 수정부(1117)는 도 7의 제공된 코드 영역(1220)의 명령어를 사용자가 수정할 수 있도록 할지, 수정하지 못하도록 할지 선택하는 부분이다. 명령어 추가부(1118)는 도 7의 제공된 코드 영역(1220)의 명령어 외에 사용자가 명령어를 추가할 수 있도록 할지 선택하는 부분이다.

[0050] 코드 제공 영역(1120)은 제공할 코드와 설명글 입력부(1121), 제공할 코드 목록부(1122), 제공할 코드의 텍스트 형태 목록부(1123), 실행 버튼부(1124), 제공할 코드의 실행 결과부(1125), 현재 제공된 코드를 정답으로의 등록부(1126), 그리고 정답으로 등록된 코드 목록부(1127)를 포함한다.

[0051] 제공할 코드와 설명글 입력부(1121)는 도 7의 1210과 같이 다른 사용자에게 제공할 코드와 설명글을 입력하는 부분이다. 제공할 코드 목록부(1122)는 도 7의 1221과 같이 다른 사용자에게 제공할 코드의 목록을 나타낸 부분이다. 제공할 코드의 텍스트 형태 목록부(1123)는 제공할 코드 목록부(1122)를 텍스트 형태로 보여주는 목록이다.

[0052] 실행 버튼부(1124)는 제공할 코드 목록부(1122)에 조합한 명령어를 실행하기 위한 버튼이다. 제공할 코드의 실행 결과부(1125)는 제공할 코드 목록부(1122)를 실행한 결과가 나타나는 부분이다. 현재 제공된 코드를 정답으로의 등록부(1126)는 제공할 코드 목록부(1122)의 명령어를 정답 코드로 등록하는 버튼이다. 그리고 정답으로 등록된 코드 목록 확인 버튼(1127)은 정답으로 등록된 코드의 목록을 확인하는 버튼이다. 정답으로 등록된 코드는 도 14와 같이 조합된 코드의 정답률을 비교할 때 사용된다.

[0053] 문제 등록부(1100)는 학습과 평가를 동시에 수행하는데 필요한 학습 자료를 제공하는 역할을 한다. 즉, 문제의 내용과 문제를 해결하기 위한 코드를 제공한다. 문제 등록자는 코드의 제공 형태를 선택할 수 있다.

[0054] 코드의 제공 형태는 첫째, 문제 등록자가 입문자에게 제공한 코드를 무작위로 제공할지 여부이다. 둘째, 문제 등록자가 입문자에게 제공한 코드를 입문자가 수정할 수 있도록 할 지 여부이다. 셋째, 문제 등록자가 입문자에게 제공한 코드 외에, 입문자가 코드를 추가할 수 있도록 할 지 여부이다. 일한 문제라도 코드의 제공 형태에 따라 난이도가 달라질 수 있다.

[0055] 가장 쉬운 1단계인 따라하기(Follow Way)는 파이썬 코드를 학습하기 위한 것으로 문제 등록자가 등록한 순서대로 입문자에게 코드를 제공한다. 프로그램의 순서, 들여쓰기 등의 모든 절차가 완성된 형태로 제공하여 입문자는 제공된 코드를 그대로 따라서 옮기면서 코드를 학습하고 프로그램을 완성할 수 있다.

[0056] 2단계인 배치하기(Arrange Blocks)는 문제 풀이 절차를 학습하기 위한 것으로 문제 등록자가 제공한 코드의 순서를 무작위로 섞어서 입문자에게 제공하고, 들여쓰기를 해제하여 입문자가 스스로 문제 풀이 절차를 찾을 수 있도록 하는 형태이다.

[0057] 3단계인 수정하기(Modify Blocks)는 코드를 수정할 수 있도록 하기 위한 것으로, 코드 등록자가 입문자에게 제공한 코드 중 잘못된 부분을 입문자가 수정하는 형태이다.

[0058] 4단계인 추가하기(Add Blocks)는 코드를 추가할 수 있도록 하기 위한 것으로, 코드 등록자가 입문자에게 제공한 코드 외에, 프로그램을 완성하기 위해 입문자가 일부 코드를 직접 추가하도록 하는 형태이다.

[0059] 도 7은 도 5에 도시된 프로그래밍 활동부를 예시적으로 보여주는 화면이다. 도 7을 참조하면, 프로그래밍 활동부(1200)는 제공된 문제 영역(1210), 제공된 코드 영역(1220), 코드 조합 영역(1230), 코드 실행 결과 영역(1240), 텍스트 코드 영역(1250)을 포함한다. 각 영역에 대한 설명은 다음과 같다.

[0060] 제공된 문제 영역(1210)은 문제 등록자가 제공한 문제 내용을 입문자가 확인하는 영역이다. 제공된 문제 영역(1210)에는 문제의 힌트 영역(1211)이 포함되어 있다. 문제의 힌트 영역(1211)은 도 6의 문제의 힌트 입력부(1115)에 입력한 내용이 표시된다.

[0061] 제공된 코드 영역(1220)은 문제 등록자가 제공한 명령어를 입문자가 확인하는 영역이다. 프로그래밍에 대한 능숙함의 단계를 고려하여 교수자가 설정해 둔 4단계인 따라하기, 배치하기, 수정하기, 추가하기 형태로 프로그래밍 학습이 이루어진다. '따라하기'는 연습을 위한 단계이고, '배치하기'는 주어진 명령어의 순서를 맞추는 것이다. '수정하기'는 제공된 명령어의 일부를 입문자가 직접 수정해서 프로그래밍하는 단계이며, '추가하기'는 입문자가 제공된 명령어 이외에 필요하다고 판단되는 명령어를 직접 만들어서 구현하는 단계이다.

[0062] 제공된 코드 영역(1220)은 제공된 명령어 목록부(1221), 제공된 명령어의 설명글 확인부(1222), 제공된 명령어의 수정 부(1223), 제공된 명령어의 수정 완료부(1224), 제공된 명령어의 삭제부(1225), 제공된 명령어의 한 줄씩 실행부(1226), 제공된 명령어의 전체 실행부(1227), 제공된 명령어와 정답으로 등록된 코드와 비교하기 위한 코드 비교부(1228), 그리고 제공된 명령어를 초기화 하기 위한 명령어 초기화부(1229)를 포함한다.

[0063] 제공된 명령어 목록부(1221)는 문제를 만든 사용자가 만든 도 6의 제공할 코드 목록부(1126)와 동일한 목록이다. 문제를 만든 사용자가 도 6의 1116에서 제공할 명령어의 순서를 "무작위(Random)"로 선택한 경우, 순서가 무작위로 제공되며, "허용안함(Deny)"를 선택한 경우, 도 6의 제공할 코드 목록부(1126)와 동일한 목록이 표시된다.

[0064] 제공된 명령어의 설명글 확인부(1222)는 문제를 만든 사용자가 도 6의 문제의 힌트 입력부(1115)에 입력한 내용이 표시된다. 제공된 명령어의 수정부(1223)는 제공된 명령어를 수정할 때 사용되는 부분이며 버튼을 누르도록 설정될 수 있다. 제공된 명령어의 수정 완료부(1224)는 제공된 명령어의 코드와 설명글을 저장할 때 사용되는 부분이며 버튼을 누르도록 설정될 수 있다. 제공된 명령어의 삭제부(1225)는 제공된 명령어를 삭제할 때 사용되는 부분이며 버튼을 누르도록 설정될 수 있다.

[0065] 제공된 명령어의 한 줄씩 실행부(1226)는 제공된 명령어를 코드 조합 영역(1230)으로 이동 시킨 후 한 줄씩 실행하는 부분이며 버튼을 누르도록 설정될 수 있다. 제공된 명령어의 전체 실행부(1227)는 제공된 명령어를 코드 조합 영역(1230)으로 이동 시킨 후 한 번에 실행하는 부분이며 버튼을 누르도록 설정될 수 있다.

[0066] 코드 비교부(1228)는 제공된 명령어와 정답으로 등록된 코드를 비교할 수 있다. 코드 비교부(1228)는 제공된 명령어를 코드 조합 영역(1230)으로 이동 시킨 후 명령어와 문제를 만든 사용자가 도 6의 현재 제공된 코드를 정답으로의 등록부(1126) 버튼을 눌러 등록한 정답 코드와 비교할 때 누르는 버튼일 수 있다.

[0067] 제공된 명령어를 초기화하기 위한 명령어 초기화부(1229)는 문제를 만든 사용자가 도 6의 제공할 코드 목록부(1122)에 제공한 목록으로 돌아가기 위한 버튼일 수 있다. 문제를 만든 사용자가 도 6의 1116에서 제공할 명령어의 순서를 "무작위(Random)"로 선택한 경우, 순서가 무작위로 제공되며, "허용안함(Deny)"를 선택한 경우, 도 6의 제공할 코드 목록부(1126)와 동일한 목록이 표시된다.

[0068] 코드 조합 영역(1230)은 제공된 코드를 입문자가 조합하여 프로그래밍하는 학습 활동 영역이다. 입문자는 “코

드 제공 영역”에서 마우스로 드래그 앤 드롭 하여 “코드 조합 영역”으로 블록을 이동한다.

[0069] 코드 실행 결과 영역(1240)은 코드 조합 영역(1230)에 조합된 명령어를 제공된 명령어의 한 줄씩 실행부(1226)이나 제공된 명령어의 전체 실행부(1227)의 버튼을 눌렀을 때의 결과가 나타나는 영역이다. 텍스트 코드 영역(1250)은 코드 조합 영역1230)에 조합한 명령어가 텍스트로 조합되어 나타나는 영역이다.

[0070] 도 8 내지 도 12는 도 7에 도시된 코드 조합 영역을 예시적으로 보여주는 화면이다. 도 8을 참조하면, 코드 조합 영역(1230)은 코드 이동부(1231) 및 조합된 코드의 텍스트 형태 목록부(1232)를 포함한다.

[0071] 코드 이동부(1231)는 명령어를 이동하는 부분이다. 명령어를 드래그하면, 이동가능한 위치가 노란색으로 표시된다. 드롭하여 명령어를 내려 둘 수 있다. 조합된 코드의 텍스트 형태 목록부(1232)는 명령어를 이동하면 조합된 코드가 텍스트 형태로 표현되는 부분이다.

[0072] 도 8을 참조하면, “코드 제공 영역”에서 “코드 조합 영역”으로 코드를 드래그 앤 드롭하여 프로그래밍하는 과정을 나타낸 것으로 1단계~4단계에서 사용되는 방법이다. 옮겨진 코드가 위치할 곳에 옅은 노란색으로 표시된다. 명령어는 “코드 제공 영역”과 “코드 조합 영역” 안에서도 이동할 수 있다.

[0073] 도 9를 참조하면, 코드 조합 영역(1230)은 조합된 명령어의 전체 실행부(1233), 조합된 명령어의 실행 결과부(1234), 조합된 명령어의 수정부(1235), 수정할 명령어의 코드와 설명글 입력부(1236), 그리고 수정할 명령어의 수정 완료부(1237)를 포함한다.

[0074] 조합된 명령어의 전체 실행부(1233)는 사용자가 조합한 명령어 전체를 한 번에 실행하는 버튼일 수 있다. 조합된 명령어의 실행 결과부(1234)는 사용자가 조합한 명령어를 실행한 결과가 표시되는 부분이다. 조합된 명령어의 수정부(1235)는 조합된 명령어를 수정하기 위해 누르는 버튼일 수 있다. 문제를 만든 사용자가 도 6의 1117에서 "수정(Modify)"을 "허용(Allowed)" 한 경우에만 명령어 수정 버튼이 보이게 된다. 수정할 명령어의 코드와 설명글 입력부(1236)는 수정하고자 하는 명령어의 코드와 설명글을 입력하는 부분이다. 수정할 명령어의 수정 완료부(1237)는 수정한 명령어의 코드와 설명글을 저장하는 버튼일 수 있다.

[0075] 도 9를 참조하면, 3단계인 수정하기 예시이다. 실행 버튼을 누르면 “코드 실행 결과 영역”에서 실행 결과를 확인할 수 있다. 명령어의 수정이 필요하다고 판단되면, 톱니바퀴 모양의 아이콘을 'click'하여 스스로 수정할 수 있다. 삭제가 필요하면, 빨간색 휴지통 아이콘을 활용한다.

[0076] 도 10을 참조하면, 코드 조합 영역(1230)은 명령어 추가부(1238), 그리고 추가할 명령어의 코드와 설명글 입력부(1239)를 포함한다.

[0077] 명령어 추가부(1238)는 사용자가 명령어를 추가하고자 할 때 누르는 버튼일 수 있다. 문제를 만든 사용자가 도 6의 1118에서 "추가(Add)"를 "허용(Allowed)" 한 경우에만 명령어 추가 버튼이 보이게 된다. 추가할 명령어의 코드와 설명글 입력부(1239)는 추가하고자 하는 명령어의 코드와 설명글을 입력하는 부분이다. 명령어를 추가하면 설명글은 보이지 않고 명령어의 코드만 보이게 된다.

[0078] 도 10을 참조하면, 4단계인 추가하기 예시로, 코드 한 줄을 입문자가 직접 추가하는 것이다. '+'모양의 아이콘을 눌러 파이썬 코드와 코드에 대한 설명을 추가할 수 있다.

[0079] 도 11을 참조하면, 코드 조합 영역(1230)은 조합된 명령어의 한 줄씩 실행부(1241), 조합된 명령어의 현재 실행 위치부(1242), 그리고 조합된 명령어의 처음부터 현재까지의 실행 결과부(1243)를 포함한다. 조합된 명령어의 한 줄씩 실행부(1241)는 조합된 명령어를 한 줄씩 실행시키는 버튼일 수 있다. 조합된 명령어의 현재 실행 위치부(1242)는 조합된 명령어를 한 줄씩 실행시킬 때, 현재 실행되고 있는 위치를 보라색으로 표시한다. 조합된 명령어의 처음부터 현재까지의 실행 결과부(1243)는 조합된 명령어를 한 줄씩 실행한 결과를 표시한다.

[0080] 도 11을 참조하면, 디버깅 버튼을 'click'하여 코드를 한 줄씩 실행한 결과를 확인할 수 있다. 한 줄씩 실행되는 결과는 “코드 실행 결과 영역”에서 확인할 수 있다.

[0081] 도 12를 참조하면, 코드 조합 영역(1230)은 조합된 명령어 중 오류가 발생한 명령어의 표시부(1244), 그리고 오류 메시지부(1245)를 포함한다. 조합된 명령어 중 오류가 발생한 명령어의 표시부(1244)는 사용자가 오류의 위치를 시각적으로 인식할 수 있도록 노란색으로 표시된다. 오류 메시지부(1245)는 오류의 위치와 오류의 원인이 표시된다.

[0082] 도 12를 참조하면, 디버깅 버튼을 'click'한 경우, 실행 과정에서 오류가 발생한 경우이다. 오류가 발생하면, “코드 실행 결과 영역”에 오류 메시지가 출력된다. 실행 버튼을 'click'할 경우에는 오류가 처음 발생한 지점

이 노란색으로 표시된다.

[0083] 도 13은 프로그래밍 요소의 문제 풀이 순서를 나타낸 도면이다. 입문자는 “input, operator, if statement, list, for statement, function”으로 구성된 7개의 프로그래밍 요소에 대해 “따라하기”, “배치하기”, “수정하기”, “추가하기” 형태로 학습할 수 있다.

[0084] 1단계인 “따라하기”는 도 7의 “제공된 코드 영역”에 제시된 명령어를 “코드 조합 영역”으로 그대로 따라서 이동시키는 활동으로 가장 쉬운 유형이다. 입문자는 “따라하기” 유형을 통해 파이썬 명령어를 확인할 수 있다.

[0085] 2단계인 “배치하기”는 무작위로 배치된 코드를 올바른 순서대로 나열하는 것이며, 순서를 모르는 경우 “따라하기”에서 활동한 내용을 참고할 수 있다.

[0086] 3단계인 “수정하기”는 “따라하기”와 동일한 순서로 코드가 제공되지만, 코드의 일부 내용이 잘 못되어 있거나 프로그램 완성을 위한 코드가 누락되어 있어 입문자가 코드를 수정하거나 추가해야 한다.

[0087] 4단계인 “추가하기”는 프로그램을 완성하기 위해 입문자가 직접 코드를 추가해야 한다.

[0088] 도 14는 문제 풀이 결과를 나타낸 도면이다. 정답 확인 버튼(check the answer)을 누르면, 정답과 얼마나 일치하는지 표시되며 정답과 완전히 일치하면 100%로 표시된다.

[0089] 정답률을 계산하는 방식은 Levenshtein distance 알고리즘을 사용한다. 예로서 알고리즘은 다음과 같다.

$$\operatorname{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \operatorname{lev}_{a,b}(i-1,j) + 1 \\ \operatorname{lev}_{a,b}(i,j-1) + 1 \\ \operatorname{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

[0090]

[0091] 교사가 등록해 둔 정답과 학생이 제출한 명령어를 한 단어씩 비교하여 유사도를 계산한다.

[0092] 상술한 내용은 본 발명을 실시하기 위한 구체적인 실시 예들이다. 본 발명은 상술한 실시 예들 이외에도, 단순하게 설계 변경되거나 용이하게 변경할 수 있는 실시 예들도 포함될 것이다. 또한, 본 발명은 실시 예들을 이용하여 용이하게 변형하여 실시할 수 있는 기술들도 포함될 것이다. 따라서, 본 발명의 범위는 상술한 실시 예들에 국한되지 않고, 후술하는 특허청구범위뿐만 아니라 이 발명의 특허청구범위와 균등한 것들에 의해 정해져야 할 것이다.

부호의 설명

[0093] 1100: 문제 등록부 1110: 문제 제공 영역 1120: 코드 제공 영역

1111: 문제 제목 입력부 1112: 문제 공개 여부 선택부 1113: 문제 대표 이미지 선택부

1114: 문제 내용 입력부 1115: 문제 힌트 입력부 1116: 무작위 순서 배치부 1117: 제공된 명령어 수정부 1118: 명령어 추가부

1121: 제공할 코드와 설명글 입력부 1122: 제공할 코드 목록부 1123: 제공할 코드의 텍스트 형태 목록부 1124: 실행 버튼부 1125: 제공할 코드의 실행 결과부 1126: 현재 제공된 코드를 정답으로의 등록부 1127: 정답으로 등록된 코드 목록부

1200: 프로그래밍 활동부 1210: 제공된 문제 영역 1220: 제공된 코드 영역 1230: 코드 조합 영역 1240: 코드 실행 결과 영역 1250: 텍스트 코드 영역

1221: 제공된 명령어 목록부 1222: 제공된 명령어의 설명글 확인부 1223: 제공된 명령어의 수정부 1224: 제공된 명령어의 수정 완료부 1225: 제공된 명령어의 삭제부 1226: 제공된 명령어의 한 줄씩 실행부 1227: 제공된 명령어의 전체 실행부 1228: 코드 비교부 1229: 명령어 초기화부

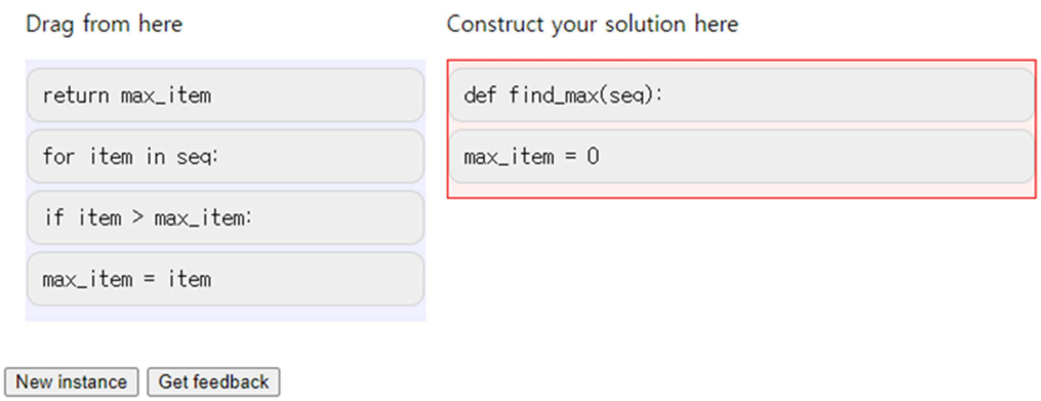
1231: 코드 이동부 1232: 텍스트 형태 목록부 1233: 조합된 명령어의 전체 실행부 1234: 조합된 명령어의 실행 결과부 1235: 조합된 명령어의 수정부 1236: 수정할 명령어의 코드와 설명글 입력부 1237: 수정할 명령어의 수정 완료부 1238: 명령어 추가부 1239: 추가할 명령어의 코드와 설명글 입력부

도면

도면1



도면2



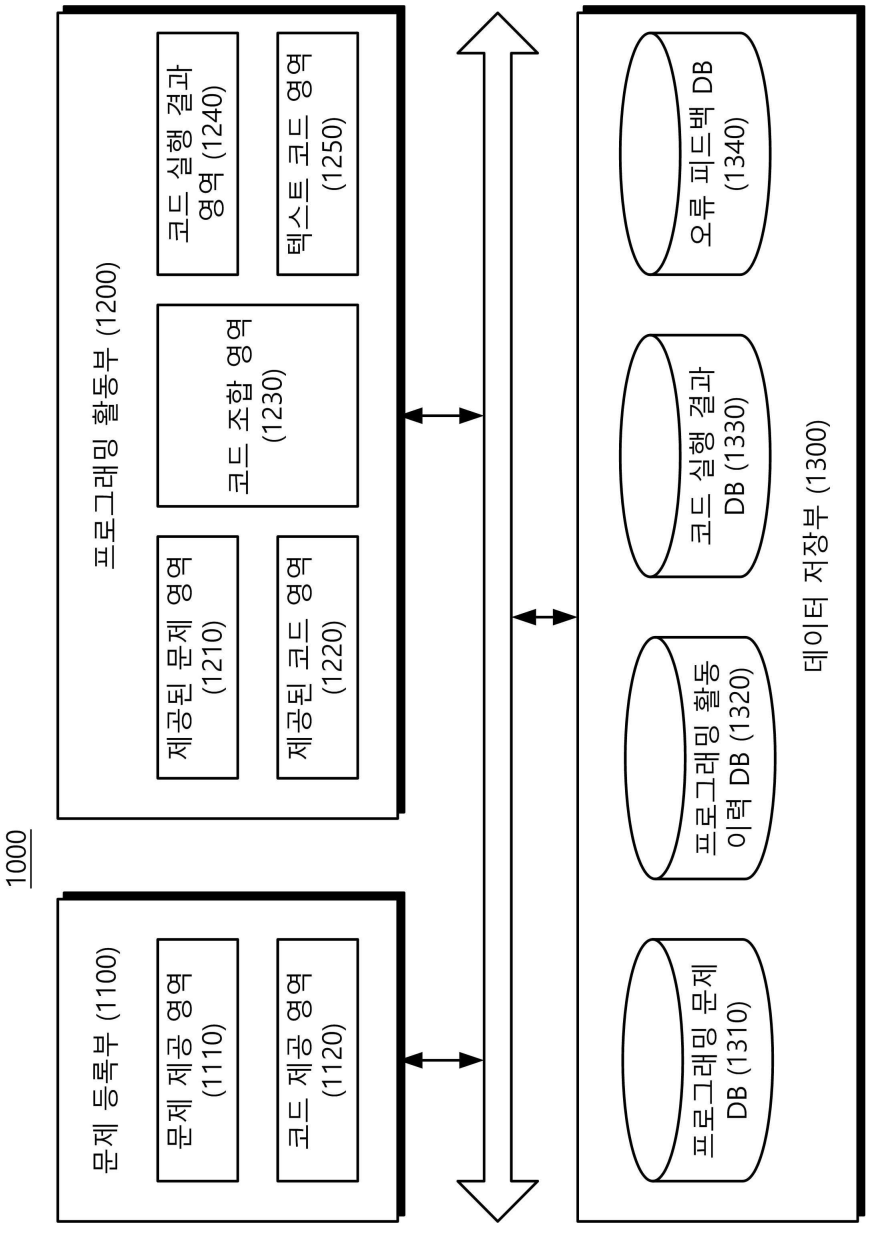
도면3

```
begin
   minimum := a[1];
   sum := ..........;
   for i from 2 to n do
   begin
      sum := sum + ..........;
      if a[i] ... min then
      begin
         minimum := ..........;
         sum := ..........;
      end;
   end;
end.
```

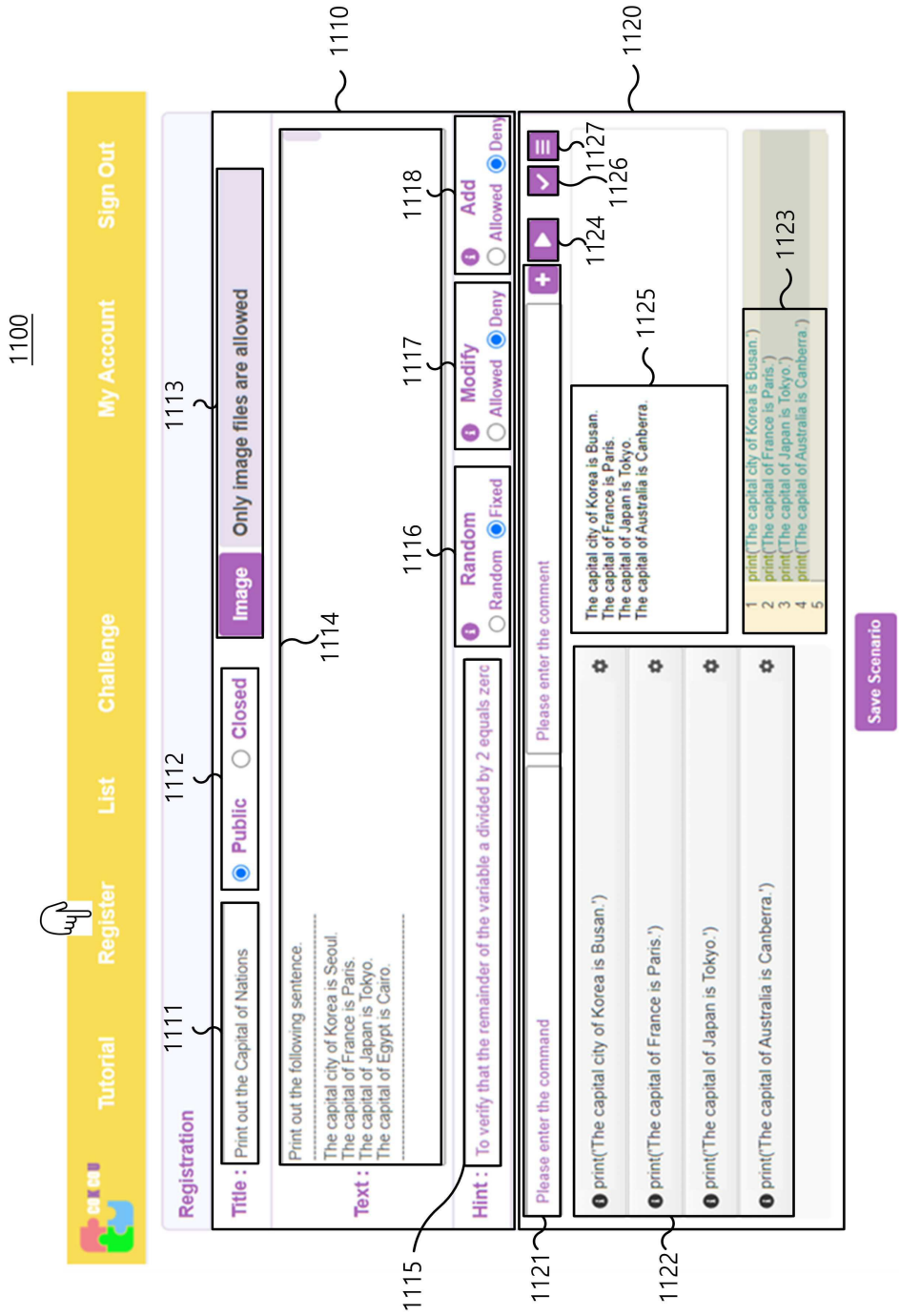
도면4

```
begin
   sum := ..... ;                                    number > 0
   read(p);                                          read(number)
   ..... ;                                           sum + number
   while i ≤ p do                                    0
   begin                                             i := i + 1
      ..... ;                                        i := 1
      if ..... then
      begin
         sum := ..... ;
      end;
      ..... ;
   end;
   write("Sum of positive integers is ",sum )
end.
```

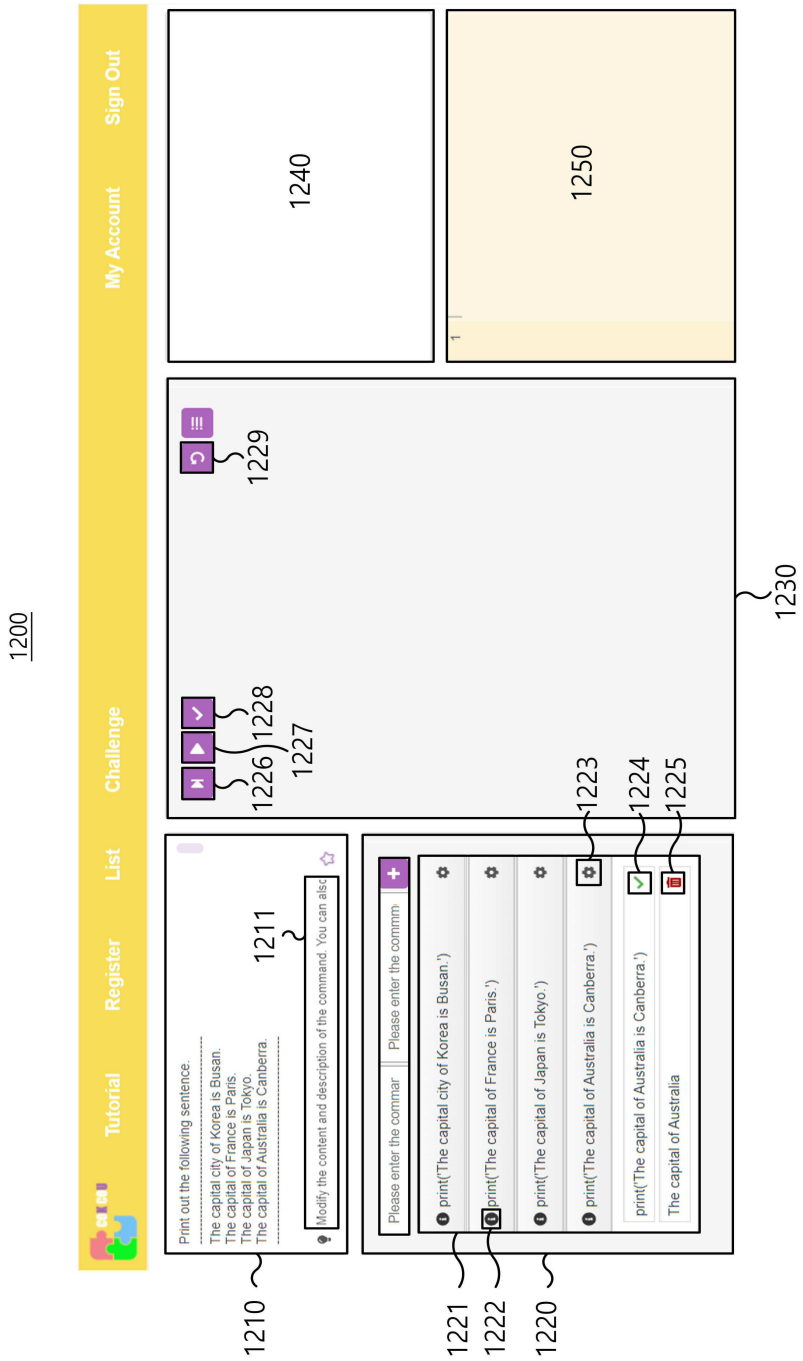
도면5



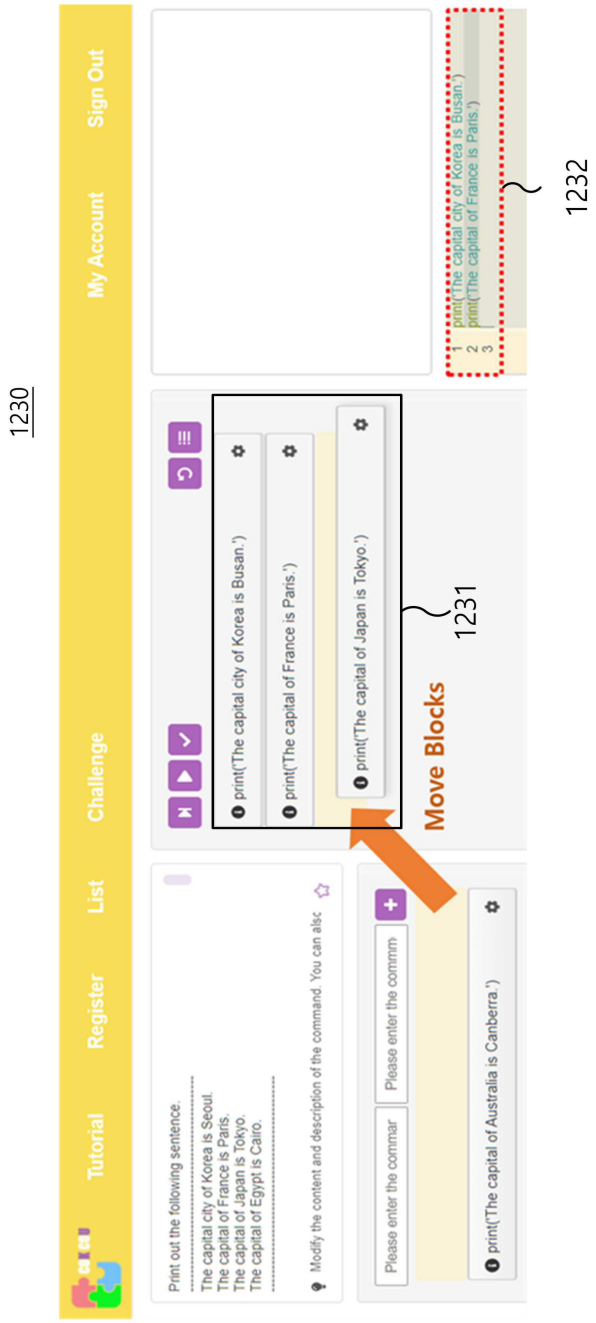
도면6


1100
Tutorial
Register
List
Challenge
My Account
Sign Out
Registration
Title :
Print out the Capital of Nations
Public
Closed
Image
Only image files are allowed
Print out the following sentence.
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Egypt is Cairo.
Text :
Hint :
To verify that the remainder of the variable a divided by 2 equals zero
Random
Fixed
Modify
Allowed
Deny
Add
Allowed
Deny
Please enter the command
Please enter the comment
print('The capital city of Korea is Busan.')
print('The capital of France is Paris.')
print('The capital of Japan is Tokyo.')
print('The capital of Australia is Canberra.')
The capital city of Korea is Busan.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Australia is Canberra.
Save Scenario
1110
1111
1112
1113
1114
1115
1116
1117
1118
1120
1121
1122
1123
1124
1125
1126
1127

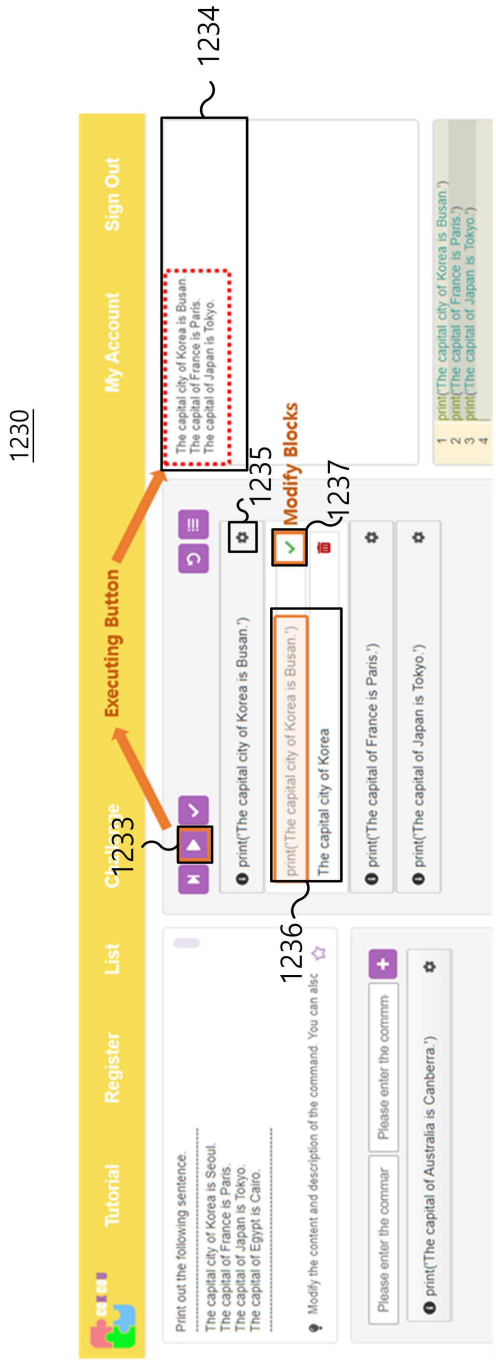
도면7


1200
Tutorial
Register
List
Challenge
My Account
Sign Out
Print out the following sentence.
The capital city of Korea is Busan.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Australia is Canberra.
Modify the content and description of the command. You can also
1210
1211
1226
1227
1228
1229
1230
1240
1250
Please enter the commar
Please enter the commm
print('The capital city of Korea is Busan.')
print('The capital of France is Paris.')
print('The capital of Japan is Tokyo.')
print('The capital of Australia is Canberra.')
print('The capital of Australia is Canberra.')
The capital of Australia
1221
1222
1220
1223
1224
1225

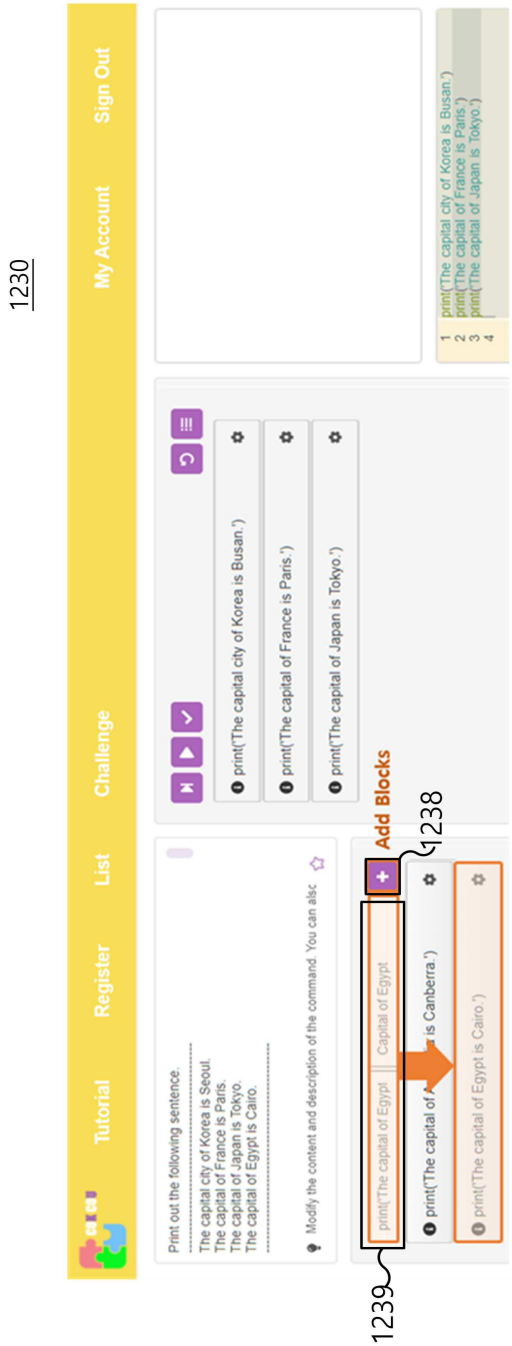
도면8


1230
Tutorial
Register
List
Challenge
My Account
Sign Out
Print out the following sentence.
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Egypt is Cairo.
Modify the content and description of the command. You can also
print('The capital city of Korea is Busan.')
print('The capital of France is Paris.')
print('The capital of Japan is Tokyo.')
Move Blocks
1231
Please enter the commar
Please enter the commm
print('The capital of Australia is Canberra.')
1 print('The capital city of Korea is Busan.')
2 print('The capital of France is Paris.')
3
1232

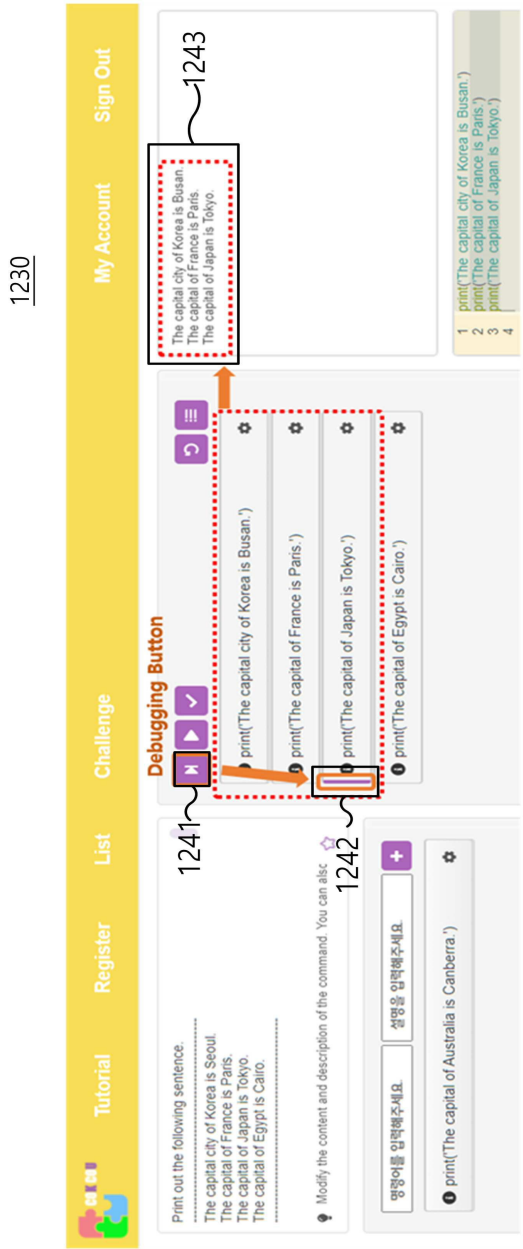
도면9


1230
1234
1235
1237
1236
1233
Executing Button
Modify Blocks
Tutorial
Register
List
My Account
Sign Out
Print out the following sentence.
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Egypt is Cairo.
print('The capital city of Korea is Busan.')
print('The capital of France is Paris.')
print('The capital of Japan is Tokyo.')
The capital city of Korea
print('The capital of Australia is Canberra.')

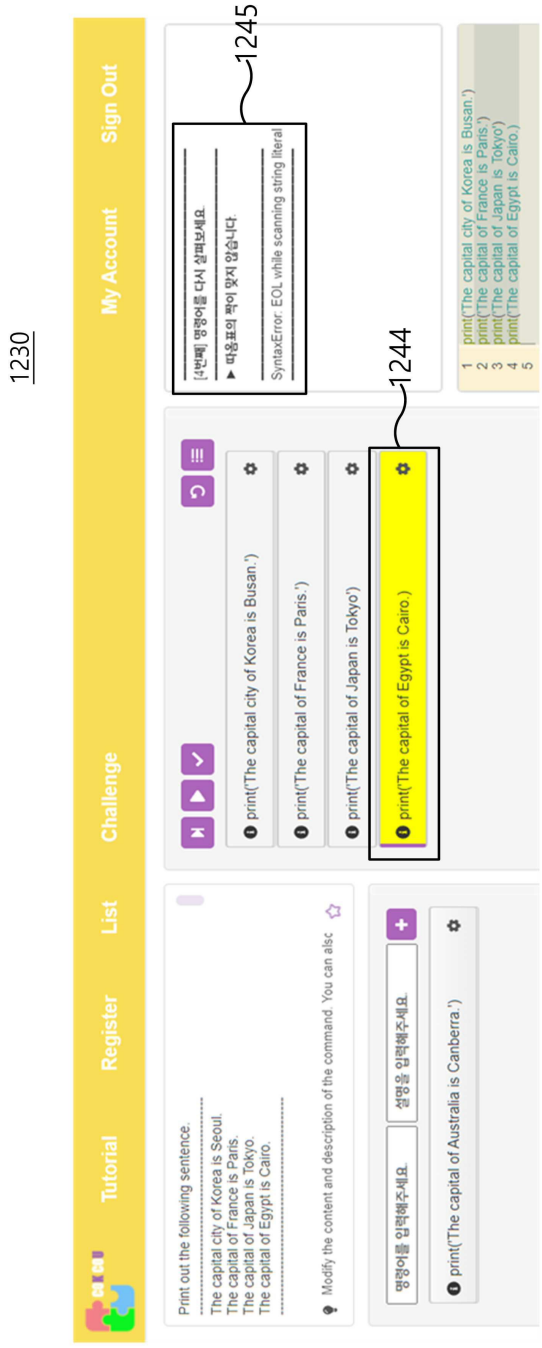
도면10


1230
Tutorial
Register
List
Challenge
My Account
Sign Out
Print out the following sentence.
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Egypt is Cairo.
Modify the content and description of the command. You can also
print('The capital city of Korea is Busan.')
print('The capital of France is Paris.')
print('The capital of Japan is Tokyo.')
Add Blocks
1238
print('The capital of Egypt | Capital of Egypt
print('The capital of Egypt is Cairo.')
1239
1 print('The capital city of Korea is Busan.')
2 print('The capital of France is Paris.')
3 print('The capital of Japan is Tokyo.')
4

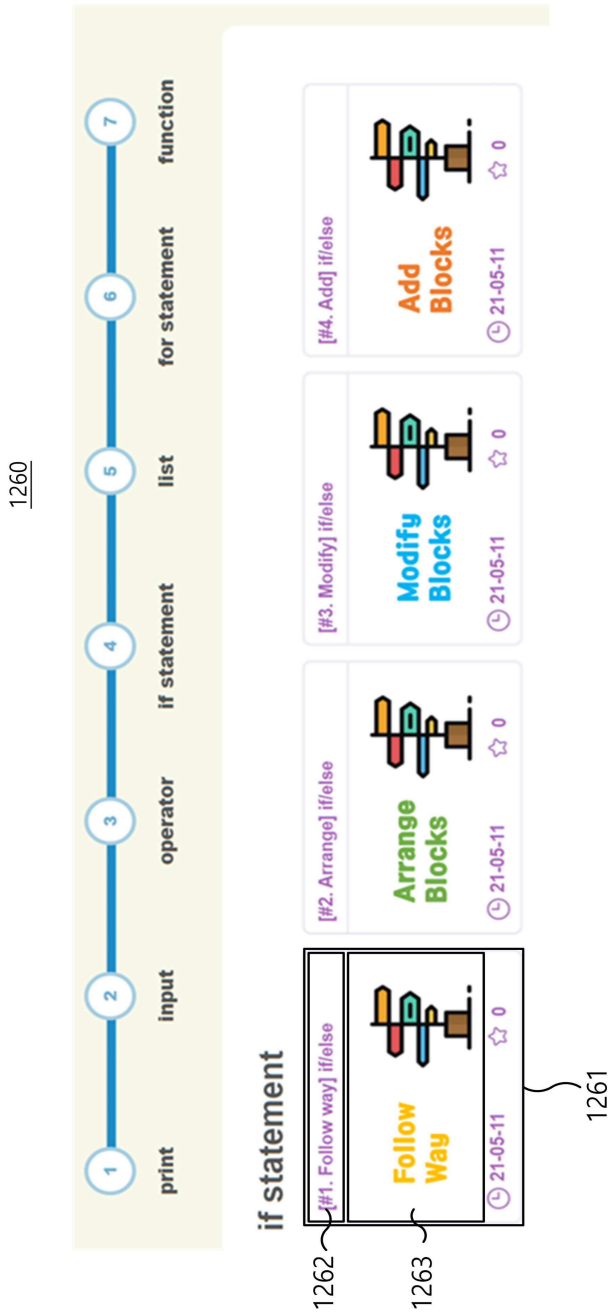
도면11


1230
Tutorial
Register
List
Challenge
My Account
Sign Out
Print out the following sentence.
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Egypt is Cairo.
Modify the content and description of the command. You can also
print('The capital of Australia is Canberra.')
Debugging Button
1241
1242
print('The capital city of Korea is Busan.')
print('The capital of France is Paris.')
print('The capital of Japan is Tokyo.')
print('The capital of Egypt is Cairo.')
1243
The capital city of Korea is Busan.
The capital of France is Paris.
The capital of Japan is Tokyo.

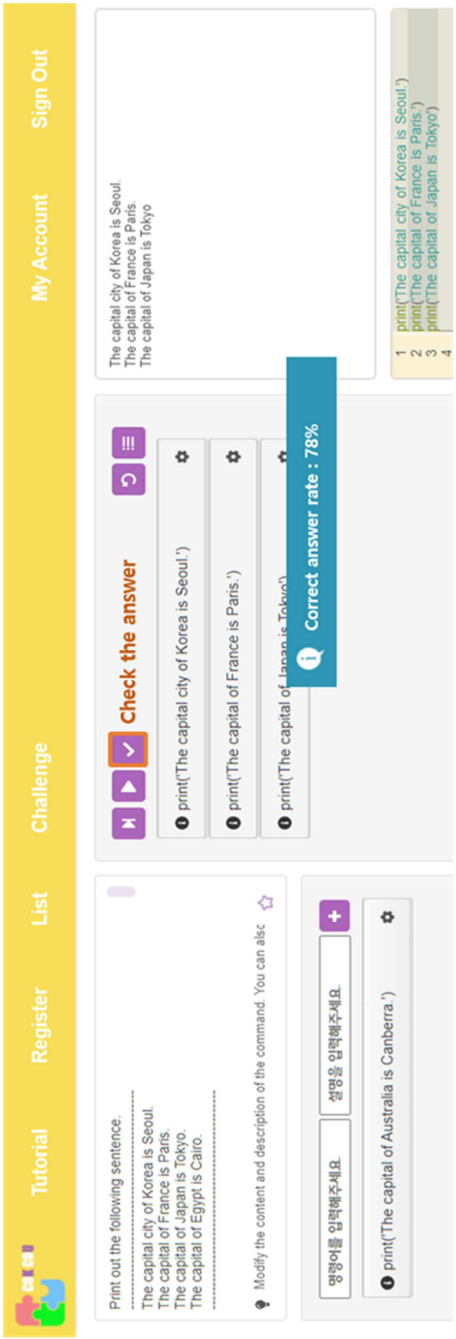
도면12


1230
Tutorial
Register
List
Challenge
My Account
Sign Out
Print out the following sentence.
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Egypt is Cairo.
Modify the content and description of the command. You can also
print('The capital city of Korea is Busan.')
print('The capital of France is Paris.')
print('The capital of Japan is Tokyo')
print('The capital of Egypt is Cairo.)
1244
[4번째] 명령어를 다시 입력보세요.
▶ 따옴표의 짝이 맞지 않습니다.
SyntaxError: EOL while scanning string literal
1245
1 print('The capital city of Korea is Busan.')
2 print('The capital of France is Paris.')
3 print('The capital of Japan is Tokyo')
4 print('The capital of Egypt is Cairo.)
5
명령어를 입력해주세요.
설명을 입력해주세요.
print('The capital of Australia is Canberra.')

도면13


1260
print
input
operator
if statement
list
for statement
function
if statement
[#1. Follow way] if/else
Follow Way
21-05-11
0
[#2. Arrange] if/else
Arrange Blocks
21-05-11
0
[#3. Modify] if/else
Modify Blocks
21-05-11
0
[#4. Add] if/else
Add Blocks
21-05-11
0
1261
1262
1263

도면14


Tutorial
Register
List
Challenge
My Account
Sign Out
Print out the following sentence.
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo.
The capital of Egypt is Cairo.
Modify the content and description of the command. You can also
명령어를 입력해주세요.
설명을 입력해주세요.
print('The capital of Australia is Canberra.')
Check the answer
print('The capital city of Korea is Seoul.')
print('The capital of France is Paris.')
print('The capital of
Correct answer rate : 78%
The capital city of Korea is Seoul.
The capital of France is Paris.
The capital of Japan is Tokyo
1 print('The capital city of Korea is Seoul.')
2 print('The capital of France is Paris.')
3 print('The capital of Japan is Tokyo')
4