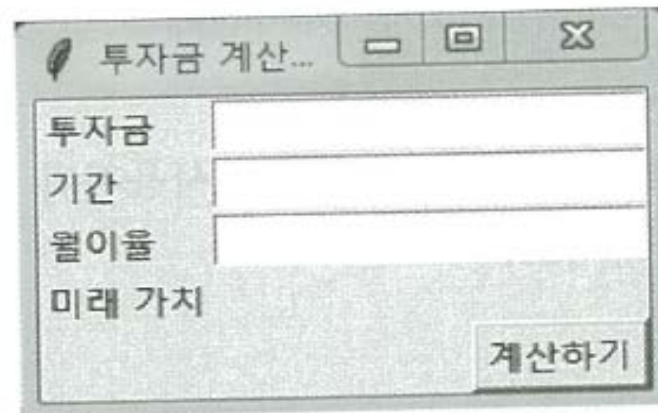

숙제5 - Tkinter



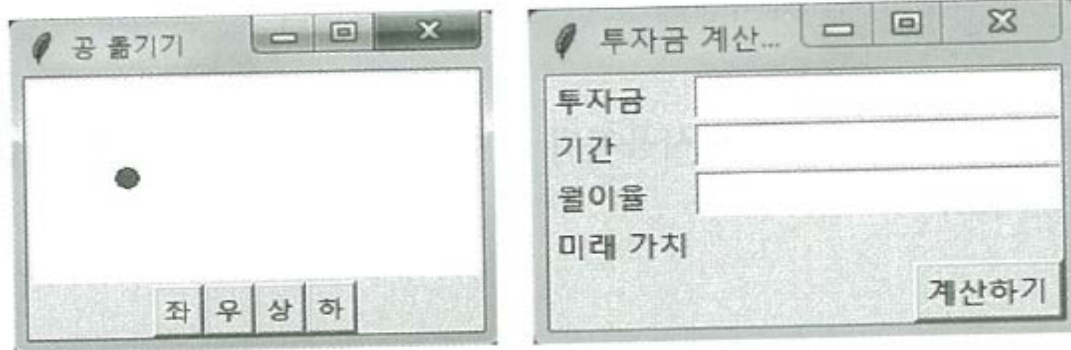
(a)



(b)

[그림 9.22] (a) 볼을 옮기기 위해 버튼을 클릭한다. (b) 투자금, 기간과 연이율을 입력하여 미래 가치를 구할 수 있다.

***9.1** (공 옮기기) 패널 내부에서 공이 움직이는 프로그램을 작성하시오. 그림 9.22(a)와 같이 공을 출력하도록 패널 클래스를 정의하고 반드시 공을 상, 하, 좌, 우로 움직이게 하기 위한 메소드를 정의하시오. 공이 화면에서 완전히 사라지는 것을 방지하기 위해 경계를 검사하시오.



(a)

(b)

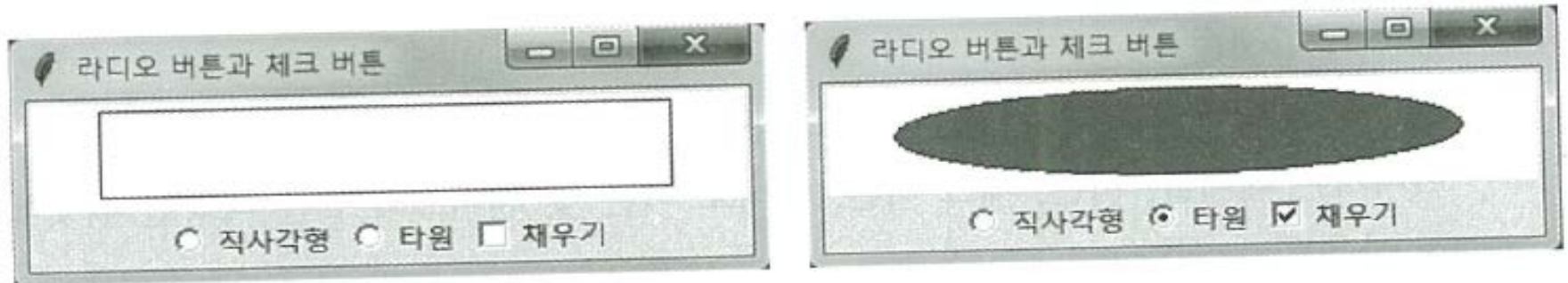
[그림 9.22] (a) 볼을 옮기기 위해 버튼을 클릭한다. (b) 투자금, 기간과 연이율을 입력하여 미래 가치를 구할 수 있다.

***9.2** (투자금 계산기 생성하기) 특정 기간 동안 주어진 연이율에 따라 투자금의 미래 가치를 계산하는 프로그램을 작성하시오. 미래 가치를 계산하는 공식은 다음과 같다.

$$\text{미래 가치} = \text{투자금} * (1 + \text{월이율})^{\text{기간} * 12}$$

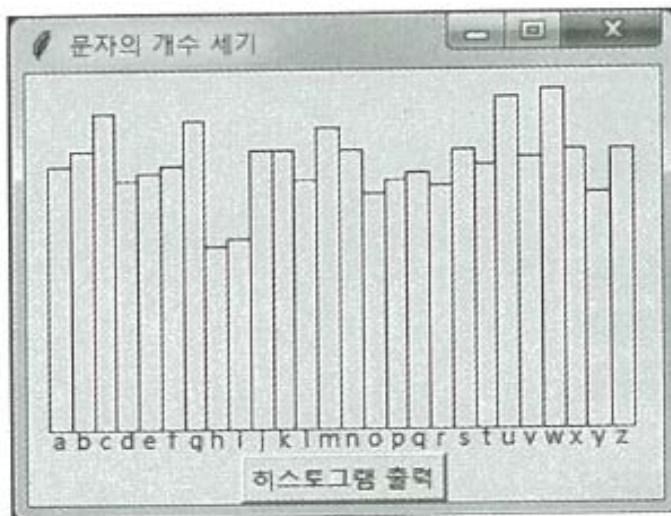
사용자가 투자금, 기간과 연이율을 입력하도록 텍스트 필드를 사용하시오. 그림 9.22(b)와 같이 사용자가 **계산하기** 버튼을 클릭하면, 텍스트 필드에 미래 가치가 출력된다.

***9.3** (기하도형 선택하기) 그림 9.23과 같이 직사각형 또는 타원을 그리는 프로그램을 작성하시오. 사용자는 라디오 버튼으로 도형을 선택하고 체크 버튼으로 도형을 채울 것인지 결정한다.

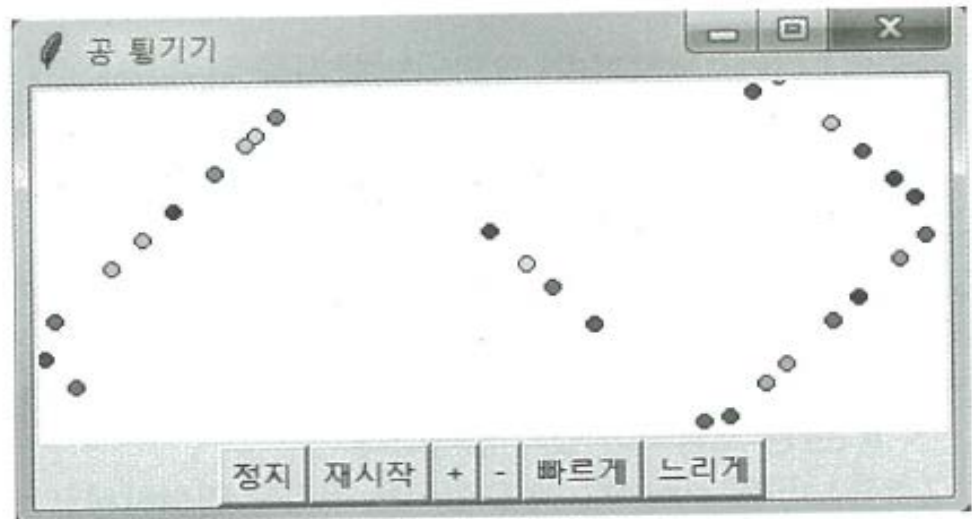


[그림 9.23] 프로그램은 사용자가 도형타입과 채움을 선택할 때 직사각형 또는 사각형을 출력한다.

10.33 (Tkinter: 막대그래프) 그림 10.16(a)와 같이 소문자 1,000개를 랜덤하게 생성하고, 각 문자의 개수를 센 후에 그 결과를 막대그래프로 출력하는 프로그램을 작성하시오.



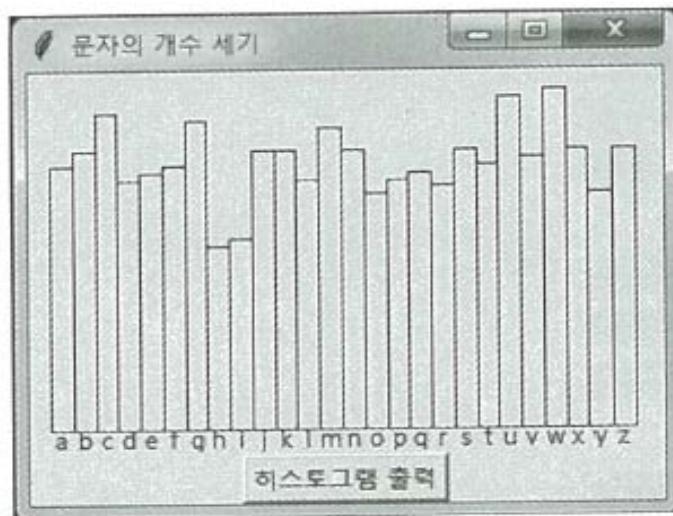
(a)



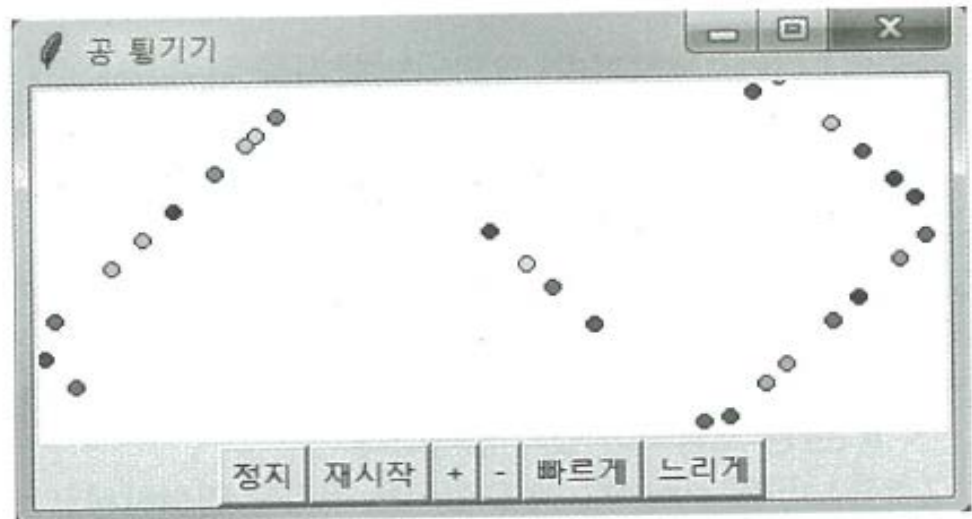
(b)

[그림 10.16] (a) 막대그래프는 각 문자의 개수를 나타낸다. (b) 공의 이동 속도를 조절하기 위해 두 버튼이 추가된다.

***10.35** (Tkinter: 공 튕기기) 그림 10.16(b)와 같이 공의 이동 속도를 빠르게 하거나 느리게 하도록 코드 10.13에 *빠르게*와 *느리게* 버튼을 추가하여 재작성하시오.

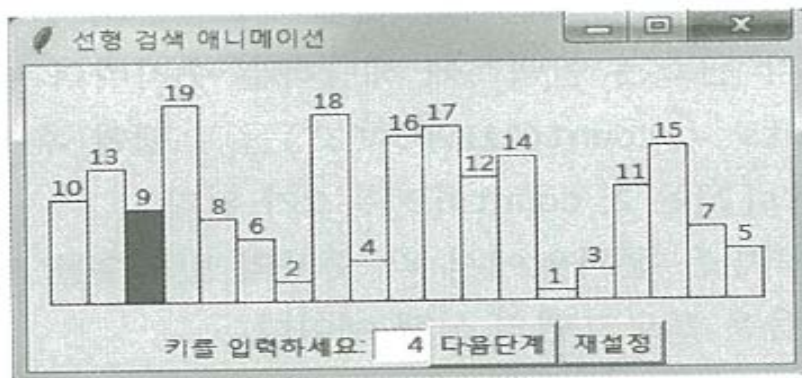


(a)

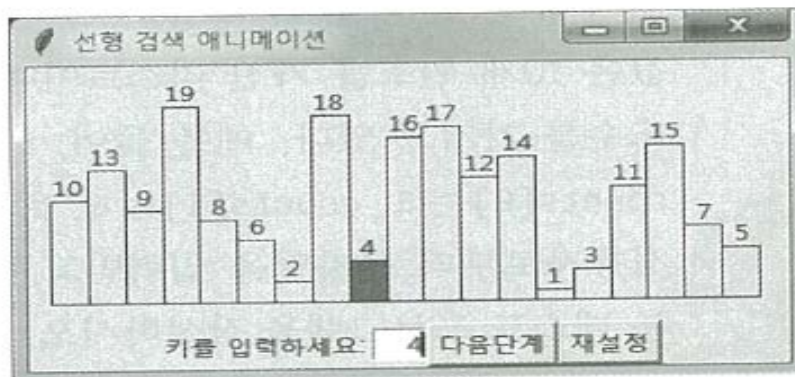


(b)

[그림 10.16] (a) 막대그래프는 각 문자의 개수를 나타낸다. (b) 공의 이동 속도를 조절하기 위해 두 버튼이 추가된다.



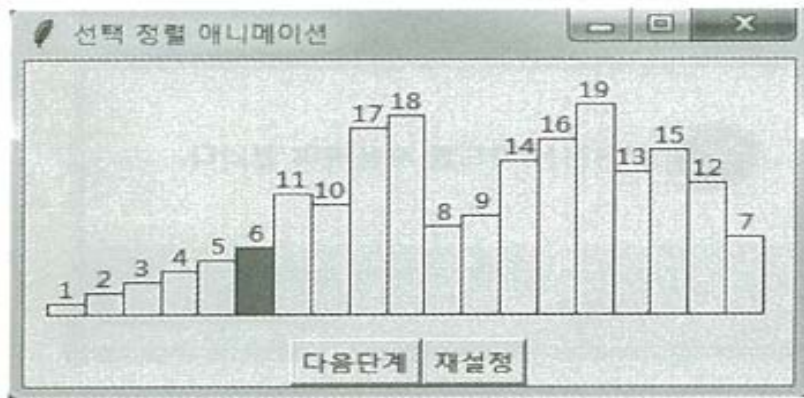
(a)



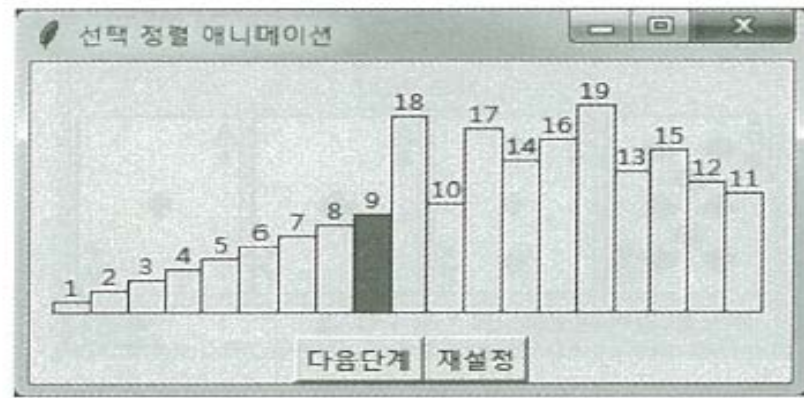
(b)

[그림 10.17] 이 프로그램은 선형 검색 알고리즘의 동작 과정을 애니메이션으로 나타낸다.

****10.36** (Tkinter: 선형 검색 애니메이션) 선형 검색 알고리즘의 동작 과정을 애니메이션으로 나타내는 프로그램을 작성하시오. 1부터 20까지의 숫자를 중복 없이 랜덤한 순서로 생성하고 이를 리스트에 저장한다. 그림 10.17에서 볼 수 있듯이, 각 원소를 하나의 막대그래프로 나타낸다. 텍스트 필드에 검색 키를 입력하도록 한다. 다음단계 버튼을 클릭하면, 프로그램은 선형 검색 알고리즘의 동작 과정에서 한 번의 비교를 수행하고 검색 위치를 나타내는 막대와 함께 막대그래프를 다시 그린다. 선형 검색 알고리즘의 동작이 종료되면 이를 사용자에게 알리기 위해 대화 상자를 출력한다. 재설정 버튼을 클릭하면, 새로운 시작을 위해 새로운 리스트를 생성한다.



(a)

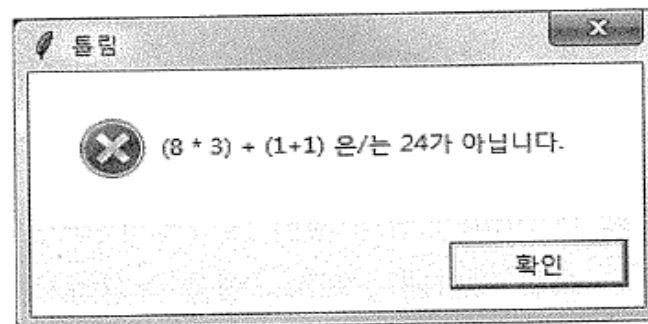
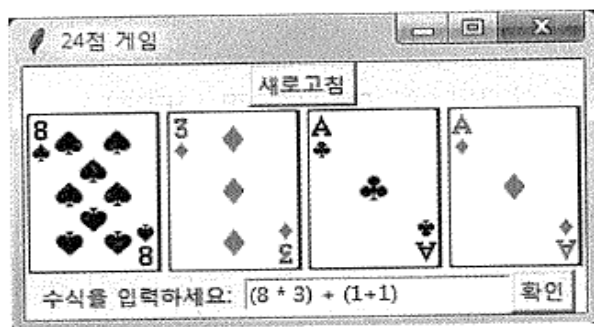
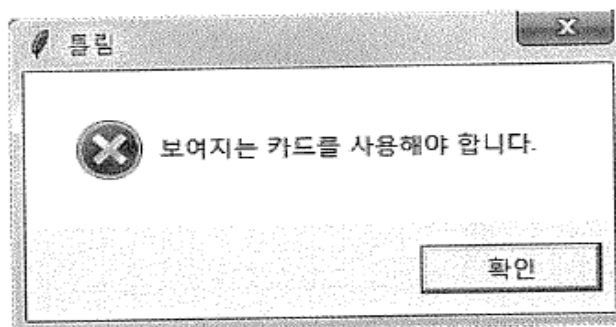
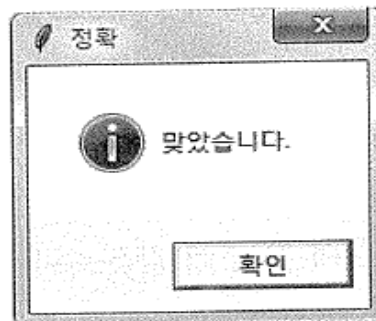
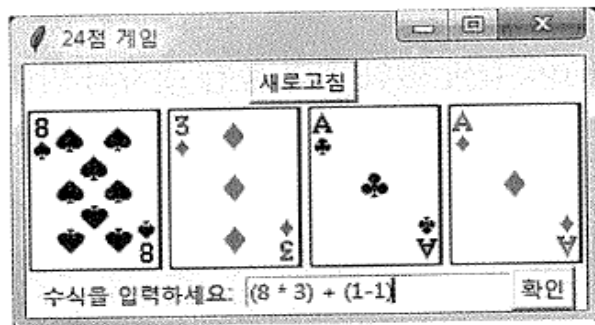


(b)

[그림 10.19] 이 프로그램은 선택 정렬 알고리즘의 동작 과정을 애니메이션으로 나타낸다.

***10.38** (Tkinter: 선택 정렬 애니메이션) 선택 정렬 알고리즘의 동작 과정을 애니메이션으로 나타내는 프로그램을 작성하시오. 1부터 20까지의 숫자를 중복 없이 랜덤한 순서로 생성하고 이를 리스트에 저장한다. 그림 10.19에서 볼 수 있듯이, 리스트의 각 원소를 막대그래프로 나타낸다. *다음단계* 버튼을 클릭하면, 프로그램은 선택 정렬 알고리즘에서 외부 루프의 반복 하나를 수행하며 새로운 리스트에 대해 막대그래프를 다시 그린다. 또한 정렬된 부분 리스트의 마지막 막대를 색칠한다. 알고리즘의 수행이 종료되면, 이를 사용자에게 알리기 위해 대화상자를 출력한다. *재설정* 버튼을 클릭하면, 새로운 시작을 위해 새로운 랜덤 리스트를 생성한다.

***10.39** (Tkinter: 24점 카드 게임) 그림 10.20과 같이 24점 카드 게임은 52장의 카드로 구성된 카드팩에서 4장의 카드를 뽑는 것부터 시작된다. 조커는 제외된다는 점에 주의하자. 각각의 카드는 숫자를 의미한다. A, K, Q, J는 각각 1, 13, 12, 11로 나타낸다. 선택된 4장의 카드의 숫자들을 이용한 수식을 입력한다. 각 카드의 숫자는 오직 한 번만 수식에 나올 수 있다. 수식은 연산자(+, -, *, /)와 괄호만을 사용하여 나타낼 수 있으며, 결과 값은 반드시 24가 되어야 한다. 사용자가 수식을 입력한 후 **확인** 버튼을 클릭하면, 수식의 숫자들이 뽑혀진 카드의 숫자들과 정확히 일치하는지를 검사하고 수식의 결과 값이 24인지를 검사한다. 또한 대화상자에 검사 결과를 출력한다. **새로고침** 버튼을 클릭하면, 4장의 카드를 다시 뽑아온다. 카드 이미지는 1.gif, 2.gif, ..., 52.gif라는 파일명으로 저장되어 있으며, 순서는 스페이드, 하트, 다이아몬드, 클로버 순이다. 그래서 처음 13개의 이미지는 스페이드 1, 2, 3, ..., 13이다.



[그림 10.20] 카드 숫자를 이용하여 사용자가 수식을 입력한다.