

소프트웨어 원리

2023

알고리즘과 코딩의 이해

알고리즘 이해
알고리즘 구성
알고리즘 구현
Turtle Module 활용

01. 알고리즘의 이해

[알고리즘Algorithm]

- 「**알고리즘**은 주어진 문제를 논리적으로 해결하기 위해 필요한 절차, 방법, 명령어들을 모아놓은 것」
- 프로그래밍을 위한 알고리즘 구성 작업
 - [해결 구성] 작업의 순서 결정 → 절차와 절차의 구성(모듈화)
 - [해결 방식] 작업의 패턴 결정 → 각 절차에서의 해결 방식
- 알고리즘 구성 시 고려요소
 - [정확성] 정확한 작업 처리
 - [신속성] 원하는 시간 내에 응답
 - [효율성] 자원의 최소 사용
 - [확장성] 유지보수/업그레이드의 용이성 → 구조적



01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업

- 작업의 순서 결정

- [문제-1] 두 컵에 담긴 물 교환

- 컵 두 개(a, b)에 각각 음료가 담겨있다. (a에는 파란색 물이 b에는 빨간색 물이)

- 두 컵의 음료를 서로 바꾸어 담고 싶다.

- 빈 컵(t)은 얼마든지 사용할 수 있다.

[질문] 어떤 과정(단계)을 거쳐야 두 컵에 물을 교환할 수 있는지 설명해보세요.

답	
---	--



01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업

- 작업의 순서 결정

- [문제-2] 할인을 적용 순서

- 할인을 적용 대상은 아래와 같다.

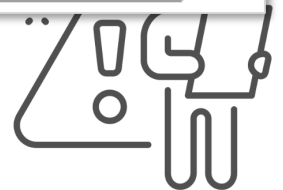
- [할인율-a] “지역주민”은 30% 할인 대상이다.

- [할인율-b] “국가유공자/현역군인”은 50% 할인 대상이다.

- [할인율-c] “70세 이상” 어르신은 100% 할인 대상이다.

[질문] 할인을 최대한 높게 결정 해주어야 한다면, 어떤 순서로 할인을 적용을 고려(검토) 하여야 가장 효율적인지 적어보시오.

답



01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업
 - 작업의 순서 결정
 - 작업의 패턴 도출

[문제-3] 할인 적용 여부 판단

○ “할인” 조건은 다음과 같다.

-[조건-A] “국가유공자”인 경우

-[조건-B] “지역 주민”인 경우

[질문] “비할인” 조건을 설명하여보시오.

답



01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업
 - 작업의 순서 결정
 - 작업의 패턴 도출

[문제-4] “합격”과 “불합격” 판단

○ “합격” 조건은 다음과 같다.

-[조건-A] 각 과목에서 40점 미만이 없어야 한다.

-[조건-B] 모든 과목의 평균 점수가 60점 이상이어야 한다.

[질문] “불합격” 조건을 설명해보세요.

답



01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업
 - 작업의 순서 결정
 - 작업의 패턴 도출

[문제-5] 요일 알아내기

○오늘은 일요일이다.

[질문] 오늘로부터 n -번째 일수가 경과한 날의 요일을 구하는 방법을 설명해보세요.

답



01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업
 - 작업의 순서 결정
 - 작업의 패턴 도출

[문제-6] 거스름 돈 지불

○ 거스름 돈 135,000원을 지폐(5만/1만/5천/1천 원권)로 지불해야 한다.

[질문] 큰 지폐 순으로 지급하여 지폐의 수를 최소화 하려고 할 때 다음 질문에 대한 식을 적어보시오. (연산자는 사칙연산, 나눈 몫, 나눈 나머지 등을 사용)

<A> 5만 원권의 개수를 구하는 식을 적어보시오.

답 1만 원권의 개수를 구하는 식을 적어보시오.

01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업

- 작업의 순서 결정

- 작업의 패턴 도출

[문제-7] 행렬 값의 순서 (I)

○ 다음 3x4열 행렬에서 값 6은 0-행, 0-열 시작 값을 기준으로 6-번째에 위치한다.
(단, 시작 값의 위치를 0-번째라고 했을 때)

	c:0	c:1	c:2	c:3
r:0	0	1	2	3
r:1	4	5	6	7
r:2	8	9	10	11

[질문] 어떤 rxc행렬(r: 행의 수, c: 열의 수)에서 임의의 i-행, j-열의 값이 위치하는 순서 값을 seq라고 했을 때, 순서 값 seq를 i와 j로부터 구하는 식을 적어보시오.
(단, 행과 열의 번호는 각각 0부터 시작한다고 가정한다.)

답

01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업
 - 작업의 순서 결정
 - 작업의 패턴 도출

[문제-8] 정n각형 그리기

○아래와 같은 작업을 4번 반복하면 정4각형 패턴을 그릴 수 있다.

- (i) 앞으로 10보 이동한다.
- (ii) 좌로 90도 회전한다.

[질문] 정n각형을 그리기 위해서 좌로 몇 도를 회전해야 하는지에 대한 일반화된 식을 적어보시오. (정확하게 n번 만에 제자리로 돌아오기 위한 회전 각도(angle) 산출 공식)

답



01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업

- 작업의 순서 결정

- 작업의 패턴 도출

[문제-9] 요일 순서 바꾸기

○아래는 각 요일에 대한 순서 Index 표이다. (Original Index: 월/화/수/목/금/토/일 순)

요일	월	화	수	목	금	토	일
순서 Index	0	1	2	3	4	5	6

위 표를 사용하되 일요일이 가장 앞서고(Index 0), 토요일(Index 6)이 가장 늦도록 바뀌어 적용하고 싶다. *New Index: 일(0)/월(1)/화(2)/수(3)/목(4)/금(5)/토(6) 순

[질문] 이전(Original) Index값으로부터 새로운(New) Index값을 환산하는 공식을 적으시오.
(예를 들어, 월요일(1)은 0이 되어야 하고, 토요일(5)은 6이 되어야 하며, 일요일(6)은 0이 되어야 함)

답

01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업

- 작업의 순서 결정

- 작업의 패턴 도출

[문제-10] 요일 순서 바꾸기

○아래는 각 요일에 대한 순서 Index 표이다. (Original Index: 일/월/화/수/목/금/토 순)

요일	일	월	화	수	목	금	토
순서 Index	0	1	2	3	4	5	6

위 표를 사용하되 월요일이 가장 앞서고(Index 0), 일요일(Index 6)이 가장 늦도록 바꾸어 적용하고 싶다. (New Index: 월/화/수/목/금/토/일 순)

[질문] 이전(Original) Index 값으로부터 새로운(New) Index 값을 환산하는 공식을 적어보시오. (예를 들어, 일요일(0)은 6이 되어야 하고, 월요일(1)은 0이 되어야 하며, 토요일(6)은 5가 되어야 함)

답

01. 알고리즘의 이해

[프로그래밍을 위한 알고리즘]

- 프로그래밍을 위한 알고리즘 구성 작업

[문제-11] 분할 정복 Divide and Conquer

○ 흰 구슬(W)과 검은 구슬(B)의 무게는 $[6W+4B] = [3W+6B]$ 일 때 일치한다.

○ 동전(C) 20개는 흰 구슬(W) 2개와 검은 구슬(B) 2개의 무게 합과 일치한다.



[질문] 결국 흰 구슬(W) 한 개의 무게는 동전(C) 몇 개의 무게와 같을까요?

답을 찾기까지의 과정을 적어보시오.

답



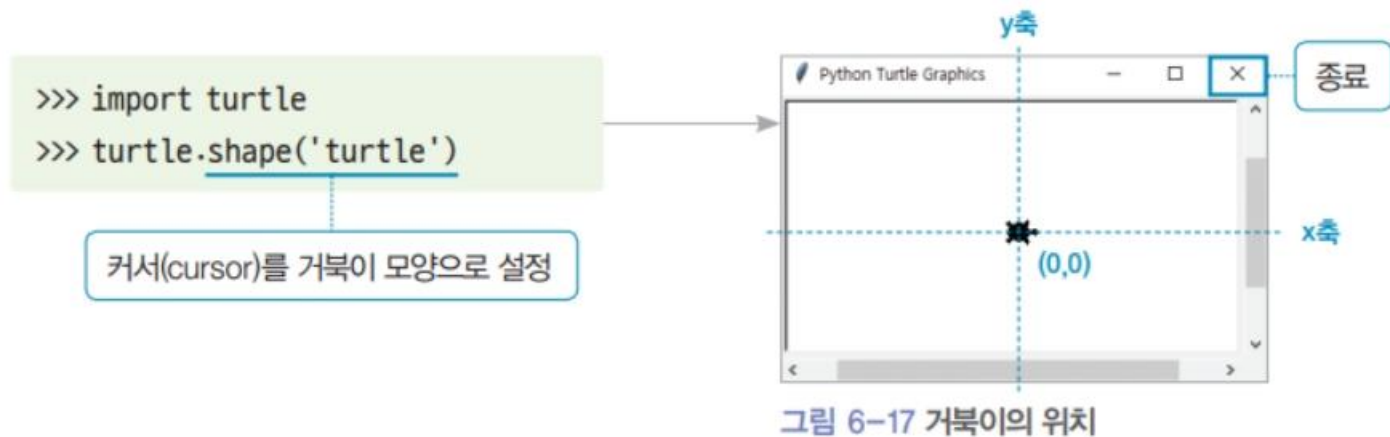
04. Turtle Module의 활용

I. 터틀 그래픽스 모듈

- 파이썬을 설치할 때 기본으로 제공되는 터틀 그래픽스라는 모듈을 이용하면 캔버스에 원하는 모양을 그리는 그래픽 프로그램을 간단하게 작성 가능

II. 터틀의 기본 사용법

- 좌표를 이용한 위치 이동뿐 아니라 픽셀pixel 단위로 거리를 지정하여 거북이를 움직일 수도 있음



04. Turtle Module의 활용

[Method 활용]

표 6-3 그리기 동작과 회전 관련 메소드

메소드명	동작	예시	인수 설명
forward(), fd()	앞으로 이동	<code>turtle.forward(100)</code> <code>turtle.fd(100)</code>	픽셀 단위의 거리
backward(), back()	뒤로 이동	<code>turtle.back(100)</code>	픽셀 단위의 거리
left(), lt()	왼쪽으로 회전	<code>turtle.left(90)</code>	회전 각도
right(), rt()	오른쪽으로 회전	<code>turtle.rt(90)</code>	회전 각도
circle()	원 그리기	<code>turtle.circle(50)</code> <code>turtle.circle(50, 180)</code>	반지름 반지름과 각도
speed()	그리기 속도 설정	<code>turtle.speed(10)</code>	0에서 10 사이의 수

04. Turtle Module의 활용

[Method 활용]

표 6-4 펜과 색상 설정 관련 메소드

메소드명	동작	예시	인수 설명
<code>pendown()</code> , <code>pd()</code> , <code>down()</code>	펜을 내려 그리기 준비	<code>turtle.pd()</code>	—
<code>penup()</code> , <code>pu()</code> , <code>up()</code>	펜을 올려 그리지 않기	<code>turtle.pu()</code>	—
<code>shape()</code>	펜 모양 설정	<code>turtle.shape('turtle')</code>	'arrow', 'circle', 'classic', 'square', 'triangle', 'turtle',
<code>pensize()</code> , <code>width()</code>	펜 두께 설정	<code>turtle.pensize(5)</code>	정수, 값이 클수록 두꺼워짐
<code>pencolor()</code>	펜 색상 설정	<code>turtle.pencolor('red')</code>	색상 이름
<code>color()</code>	펜 색상과 채우기 색상 설정	<code>turtle.color('white', 'yellow')</code>	(펜 색, 채우기 색) 순서로 색상 이름 입력
<code>begin_fill() ~ end_fill()</code>	도형의 색 채우기	<code>turtle.begin_fill()</code> <code>turtle.circle(100)</code> #원 그리기 <code>turtle.end_fill()</code>	—

04. Turtle Module의 활용

[Method 활용]

표 6-5 캔버스와 윈도우 설정 관련 메소드

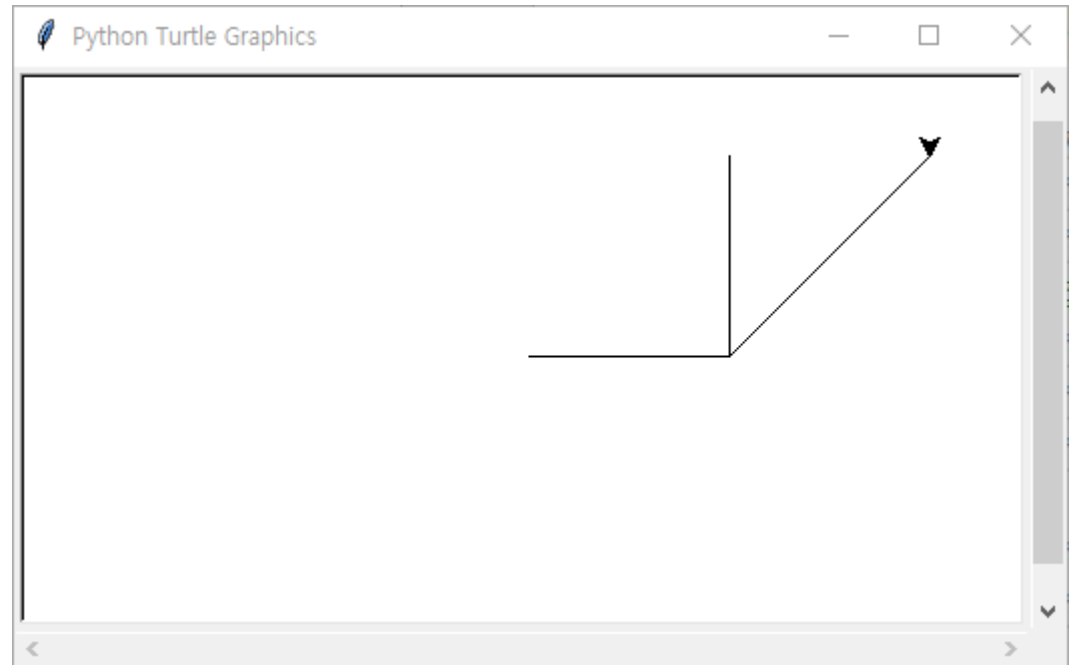
메소드명	동작	예시	인수 설명
setup()	캔버스 크기 설정	<code>turtle.setup(800, 600)</code>	폭(가로)과 높이(세로)
title()	터틀 윈도우의 제목 설정	<code>turtle.title('Painter')</code>	-
write()	캔버스에 출력하기	<code>turtle.write("ABCDEFGG")</code>	문자나 숫자 등
clear()	그림 모두 지우기	<code>turtle.clear()</code>	-
reset()	그림을 지우고, 위치와 색상 설정 초기화	<code>turtle.reset()</code>	-
done()	터틀 윈도우를 닫지 않고 그리기 종료	<code>turtle.done()</code>	-
exitonclick()	마우스를 클릭하면 터틀 윈도우 닫기	<code>turtle.exitonclick()</code>	-
bye()	터틀 윈도우 닫기	<code>turtle.bye()</code>	

04. Turtle Module의 활용

II. 터틀의 기본 사용법

➤ IDLE 실행 Window 시작 > 'IDLE' 입력 > IDLE(Python 3.10) 선택

```
>>> import turtle as t
>>> t.goto(0, 0)
>>> t.forward(100)
>>> t.right(90)
>>> t.penup()
>>> t.backward(100)
>>> t.pendown()
>>> t.forward(100)
>>> t.goto(200, 100)
```



03. 알고리즘의 구현

[프로그래밍을 위한 알고리즘 구성 실습]

◆ 정사각형 그리기

```
import turtle as t    #[Module import]
#global 변수

##### 메인 시작 #####

t.forward(100)
t.left(90)
t.forward(100)
t.left(90)
t.forward(100)
t.left(90)
t.forward(100)
t.left(90)

t.exitonclick()        # 실행 창을 닫지 않도록
##### 메인 끝 #####
```

03. 알고리즘의 구현

[프로그래밍을 위한 알고리즘 구성 실습]

◆ 정사각형 그리기

```
import turtle as t    #[Module import]
#global 변수

##### 메인 시작 #####

for _ in range(4):
    t.forward(100)
    t.left(90)

t.exitonclick()        # 실행 창을 닫지 않도록
##### 메인 끝 #####
```

03. 알고리즘의 구현

[프로그래밍을 위한 알고리즘 구성 실습]

◆ 정 n각형 그리기

```
import turtle as t    #[Module import]
#global 변수
def draw_trail(n):
    angle = 360 // n
    for _ in range(n):
        t.forward(100)
        t.left(angle)

##### 메인 시작 #####

n = int(t.numinput("변의 수 입력: ", "정수로 입력: "))
draw_trail(n)

t.exitonclick()      # 실행 창을 닫지 않도록
##### 메인 끝 #####
```

03. 알고리즘의 구현

[프로그래밍을 위한 알고리즘 구성 실습]

◆ 정 n각형 그리기

```
import turtle as t    #[Module import]
#global 변수
def draw_trail(n):
    angle = 360 // n
    for _ in range(n):
        t.forward(100)
        t.left(angle)

##### 메인 시작 #####

while True:
    n = int(t.numinput("변의 수 입력: ", "정수로 입력: "))
    draw_trail(n)

t.exitonclick()        # 실행 창을 닫지 않도록
##### 메인 끝 #####
```

03. 알고리즘의 구현

[프로그래밍을 위한 알고리즘 구성 실습]

Ex_Turtle_Angle00.py

◆ 정 n각형 그리기

```
##### 메인 시작 #####  
t.shape('turtle')  
  
while True:  
    n = int(t.numinput("변의 수 입력: ", "정수로 입력: "))  
    if n < 3:  
        if n == 0:  
            break  
        else:  
            continue  
    draw_trail(n)  
  
t.exitonclick()          # 실행 창을 닫지 않도록  
##### 메인 끝 #####
```

<https://github.com/dndxor/BCPy>

공부를 하는 이유 두 가지

1. 무엇을 하고 싶은지 알아내기 위한 것
2. 하고 싶은 바를 알아냈다면
그 것을 더 잘하는 방법을 궁리하는 것

-조창인 장편소설 '살아만 있어줘' 중에서-

Thank You !

[부천대학교]