

소프트웨어입문설계, 과제 4-2

제출기한: 2021년 3월 31일 23시 59분 (기한 내 미제출시 0점 처리)

- 본인의 *hconnect*에 생성된 본 강좌 프로젝트 (<연도>_<학수번호>_<수업코드>/<년도>_<학수번호>_<학번>.git)에 *git push*를 통해 제출된 답안만 인정함.
- 아래 예와 같은 식으로 본인의 프로젝트 아래 <과제 번호>/<문제 번호>/<각 문제의 답안 파일>의 구조가 되도록 답안 파일을 작성.

```
+ 2021_ITE1014_2021000001
+ 3-1/
+ 1/
+   - p1.py
+ 2/
+   - p2.py
+ 3/
+   - p3.py
+ ...
```

- 제출 시점은 *commit*이 작성된 시점이 아니라 *git push*가 이루어진 시점으로 판단함.
1. 정수를 하나(이하 n , 1보다 크다고 가정) 입력 받아, 1부터 n 까지의 숫자를 이용해 아래와 같은 방식으로 삼각형 모양을 출력하는 프로그램을 작성하시오. 단, 반드시 아래에 설명하는 함수를 만들고, 이 함수를 여러 번 호출해서 프로그램을 작성해야 한다. 아래 실행 예와 같이 출력을 해야한다 (↵는 사용자가 입력 후에 엔터키를 누른 것을 의미한다).
 - A. `printLine(n)`: 정수 하나를 인자로 받고(n), 1부터 인자로 받은 정수까지의 숫자를 한 줄로 출력한 후 맨 끝에서 줄바꿈 문자를 출력한다. 출력되는 각 숫자 사이에는 공백 문자가 하나씩 삽입된다. 이 함수를 한 번 호출하면 단 한 줄의 출력 결과만 출력되어야 한다.

```
(실행 예 1)

5↵
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

```
(실행 예 2)

3↵
1
1 2
1 2 3
1 2 3
1 2
1
```

- B.
- C. 제출 파일: Python 소스 파일 1개 (파일 이름은 영문으로 임의로 지정. 확장자는 **.py**)

2. 아래 제시되어 있는 함수를 구현하고, 이를 이용하여 두 개의 정수를 입력 받고 각각의 수의 약수를 구하는 프로그램을 작성하시오 (입력 받은 정수는 1보다 크다고 가정). 아래 실행 예와 같이 출력을 해야한다 (↵는 사용자가 입력 후에 엔터키를 누른 것을 의미한다).

A. problemDescription() : 아래 문구를 출력

i.

```
* Number of divisors *
```

B. getNumOfDivisors(number) : argument로 받은 number의 약수의 개수를 return 하는 함수

- i. Example)
- ii. num1 = getNumOfDivisors(12) => num1 = 6
- iii. num2 = getNumOfDivisors(6) => num2 = 4

(실행 예 1)

```
* Number of divisors *
```

```
Type the first number:
```

```
12↵
```

```
Type the second number:
```

```
6↵
```

```
Number of divisors of the first number is 6
```

```
Number of divisors of the second number is 4
```

C.

(실행 예 2)

```
* Number of divisors *
```

```
Type the first number:
```

```
5↵
```

```
Type the second number:
```

```
18↵
```

```
Number of divisors of the first number is 2
```

```
Number of divisors of the second number is 6
```

D. 제출 파일: Python 소스 파일 1개 (파일 이름은 영문으로 임의로 지정. 확장자는 **.py**)

3. 정수를 하나 입력 받아 해당 숫자만큼 아래 실행 예와 같이 다이아몬드 모양으로 별(*)을 출력하는 프로그램을 작성하시오. 단, 반드시 아래에 설명하는 함수를 만들고, 이 함수를 호출해서 프로그램을 작성해야 한다.

(실행 예 1)

```
5↵
  *
 * *
* * *
* * * *
* * * * *
* * * * *
  * * * *
    * * *
      * *
        *
```

(실행 예 2)

```
4↵
  *
 * *
* * *
* * * *
  * * * *
    * * *
      * *
        *
```

- A.
- B. `printStarDia(n)`: 정수 하나를 인자로 받아(이하 n , 1보다 크거나 같은 정수로 가정), 실행 예와 같이 다이아몬드 형태의 높이는 $n*2$ 줄, 다이아몬드 중간의 가장 두꺼운 부분에는 별(*)이 n 개 들어가는 다이아몬드 모양을 출력하는 함수.
- C. `printStarDia` 함수 안에서 반드시 중첩된 반복문(for 혹은 while)을 사용해야 한다.
- D. Hint: `print('~', end = " ")` : `print()` 함수 호출 시 줄 바꿈 하지 않음
- E. Hint: 각 줄의 별(*)을 출력하기 전에 몇 개의 공백문자를 출력하면 되는지 생각해볼 것.
- F. 제출 파일: Python 소스 파일 1개 (파일 이름은 영문으로 임의로 지정. 확장자는 **.py**)