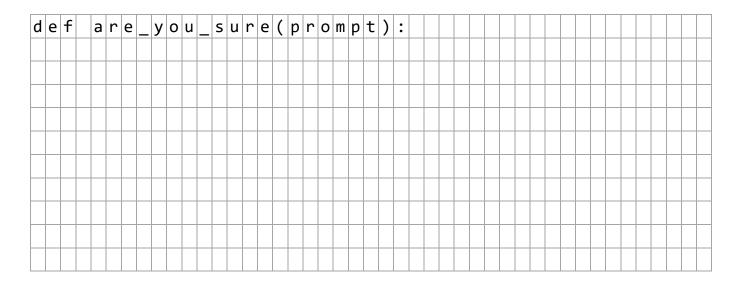
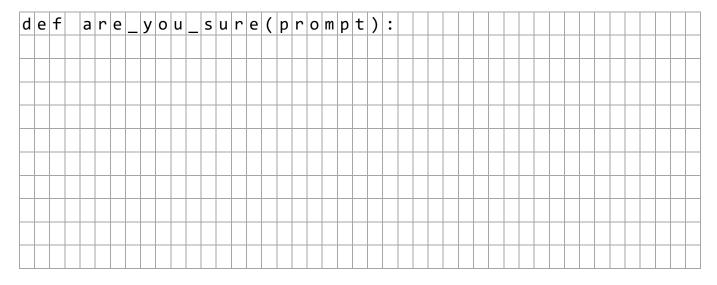
# 2016년도 2학기 컴퓨터프로그래밍기초(12729) 기말고사

																												1	학년	컨:								0	름	<u>:</u> : _							_
-	조	조급한 마음을 줄이고 한 줄 한 줄 천천히 읽어 가며 문제를 풀어 주세요.																																													
-	시험은 90분 동안 진행되며 만점은 100점입니다.																																														
1.	1 2 3	) !) !)	de 함 필 자	f 수 요	키 의 한	워 실 건	드 실행	로뱅을	人 을 중 인=	작 중 단	·하· 난힐	는 [ 통하	'힘 내는 내	}수 <del>-</del>	· 전 ore 을	d으 ak 받	'를 문 아	를 음 올	서 사 -	년 용 구	정 합 있	의 니 습	합 다. 니	니 다.	다.			Ŀ 서	<u>R</u>	(	답	1	개	, 6	점;	) -			- (	,							)
2.	1 2 3 4	) () () () ()	list list 특 nu	t는 t어 정 mb	l E	형소 내는 H H H F S	식 용 위으 라	이 물(   : 는	름 ) ) tis	입I 몇 자 t의	니디 개 list 첫	다. 있 를 턴 [	빗는 새 번째	·지  로 배 :	를 민 칸(	일 <u></u> 게	고 ·고 있-	쇼 쇼 는	[을 [을 값	<u>음</u> [ 음 [음	때 : 때 : 	는 는 가지	le ra ᅾ을	eng ing	gth ge(때	n() () 는	·를 를 nu	· 사 애mb	사용 용 per	B 합 'S	  -  -  1	디 다.  을	├. ├ /	나용	2 6	맓	.  C	<b>ŀ</b> .			(87	점)					)
(4.	. 프로그래밍 전략들 중 특히 중요한 top-down approach에 대해 30자 내외로 설명해 주세요.(8점) )  1,5) 아래 예제 코드를 읽고 물음에 답하세요.																																														
			· "		<u>"</u>						 	_		<u>"</u>		<u>.</u>	 																					$\exists$			Π	Τ	Τ	Τ			
d	e	f	-	t	h	r	e	e	(	)	:																	T				+		1					_		T		T	T			
			_	$\rightarrow$		$\vdash$	-	$\vdash$	n	÷	•	3	•																												T		T				
																																												L			
d	e	f	$\rightarrow$	$\rightarrow$		-	(	-	$\overline{}$	L													L		_		-		1		1									L	L	L	Ļ	Ļ			
	_			p	r	i	n	t		t	h	r	e	е	(	)			_				L	-		-		+		+	+	+	4	4	_	_				_	L	-	L	L	-		
n		n	(	1															_						+		+		+	+		+	+		-			$\dashv$		H	$\vdash$	$\vdash$	$\vdash$	+		-	
1	u	11	(	7																					+			+	+	+	+	+		+				$\dashv$			$\vdash$		$\vdash$	$\vdash$		-	
	위 예제 코드에 함수 정의는 총 몇 개 들어 있나요? (6점)()  F5를 눌러 위 예제 코드를 실행했을 때의 결과를 한 문장으로 설명해 주세요. (4점)																																														

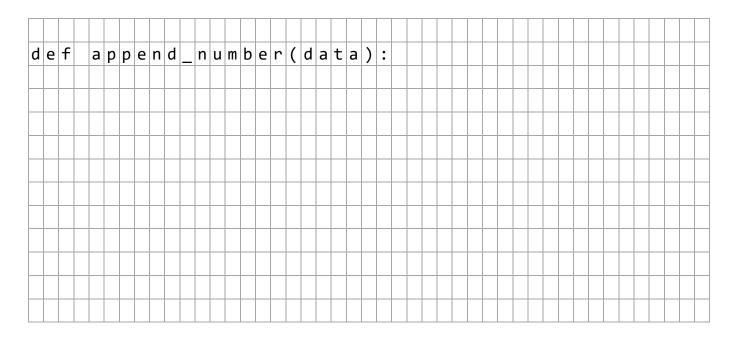
#### (6, 7, 8) 아래의 목표를 잘 읽고 각 함수의 내용물을 채워 그 함수가 목표에 맞게 실행되도록 만들어 주세요.

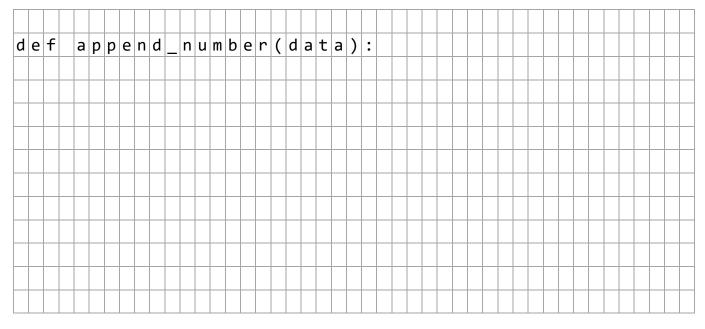
- 이 문제들을 풀 때의 주의사항:
  - 함수 내용물이 아닌 다른 코드를 적으면 0점으로 간주됩니다!
  - 모든 함수는 10개 미만의 문장으로 완성할 수 있습니다.
  - 적다가 크게 실수해서 다시 적고 싶은 경우 아래에 있는 여분의 답안 칸을 이용해 주세요. (이 때, 반드시 위 답안 칸의 내용에 가로줄을 그어 '안 쓰는 칸'임을 표기해 주세요. 그렇지 않은 경우 아래에 무엇을 적든 윗 칸의 내용으로 채점을 수행합니다)
  - 문제에서 제시하는 함수/문장 외에도 온/오프라인 수업에서 다룬 다른 함수/문장을 쓸 수 있습니다.
- 6. 사용자에게 인수 prompt에 담긴 값을 출력하고, 키 하나를 입력받은 다음, 입력한 키 값이 'y'와 같은 경우 True를 return하고, 그렇지 않은 경우 False를 return하는 함수 are\_you\_sure()를 완성해 주세요. (8점)
- 사용할 수 있는 함수:
  - write(text) 인수 text의 값을 출력합니다.
  - read\_key() 사용자가 누른 키 하나를 입력받아 해당 키 값을 return합니다.
- 사용할 수 있는 문장:
  - if문(if부분, else부분)





- 7. int 형식 값 하나를 사용자로부터 입력받고, 해당 값을 인수 data에 담긴 list의 맨 뒤에 추가하는 함수 append\_number()를 완성해 주세요.(8점)
- 사용 가능한 함수 및 수식:
  - read\_line() str 형식 값 하나를 사용자로부터 입력받아 return합니다.
  - int(x) x 값을 int 형식 값으로 바꾸는 수식입니다.
  - data.append(object) data에 담긴 list의 맨 뒤에 인수 object에 담긴 값을 추가합니다.





### 8. 아래 조건에 맞도록 함수 rank()를 완성해 주세요. (14점)

rank(data, val)

인수

data는 값들이 들어 있는 list입니다. val은 순위를 구할 값입니다.

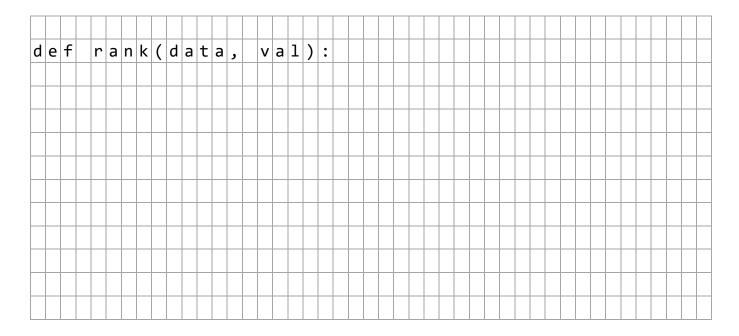
할 일

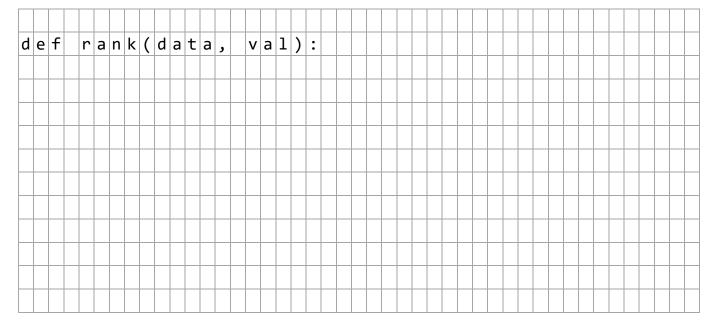
- 1. 새 변수 result에 int 형식 값 1을 담으세요.
- 2. 반복문을 사용하여 data에 있는 값 하나하나를 꺼내고, 꺼낸 그 값이 val의 값보다 더 크면 result 값에 1을 더하세요(순위 하락).

return값

변수 result에 담긴 값(방금 구한 순위 값)을 return합니다.

- 사용 가능한 문장 및 수식:
  - for문(첫 줄은 for number in data: 와 같이 적음)





### (9, 10, 11) 우리 시험의 남은 세 문제는 처음부터 직접 함수를 정의하는 문제입니다.

- 이 문제들을 풀 때의 주의사항:
  - 모든 함수는 10개 이하의 문장으로 완성**해야 합니다**. 10개보다 많은 문장을 사용한 경우 초과한 수만큼 감점이 적용됩니다.
    - ◆ 단, if문, for문, while문과 같은 복합적인 문장 자체는 세지 않습니다(간단한 문장만 셉니다).
  - 적다가 크게 실수해서 다시 적고 싶은 경우 아래에 있는 여분의 답안 칸을 이용해 주세요. (이 때, 반드시 위 답안 칸의 내용에 가로줄을 그어 '안 쓰는 칸'임을 표기해 주세요. 그렇지 않은 경우 아래에 무엇을 적든 윗 칸의 내용으로 채점을 수행합니다)
  - 문제에서 제시하는 함수/문장 외에도 온/오프라인 수업에서 다룬 다른 함수/문장을 쓸 수 있습니다.
  - (중요) 만약 문제가 어려워서 답을 적을 수 없는 경우, Python 문장 대신 주석 (#로 시작하는 한 줄)을 써서 해당 위치에서 실행해야 할 문장의 의미를 한국어 / 영어로 적어 둘 수 있습니다. 이렇게 적은 내용이 타당한 경우, 해당 문장에 대한 부분점수의 최대 절반만큼을 획득할 수 있습니다.
  - 그러니 마지막까지 힘 내 주세요!

## 9. 아래 조건에 맞도록 함수 get\_fare()을 정의해 주세요. (13점)

get\_fare(age)

인수: age는 지하철에 탑승하려는 사람의 나이 값입니다.

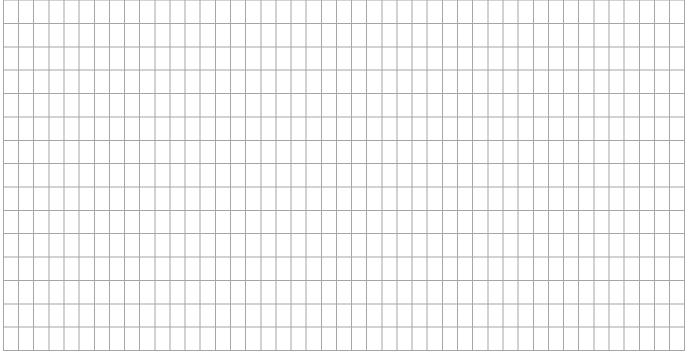
목표: 아래 표에 따라 해당 나이 값에 맞는 요금 값을 return해야 합니다:

구분(나이 범위)	요금(원)
성인(19 이상)	1250
청소년(13 이상, 19 미만)	720
어린이(13 미만)	450

### - 사용 가능한 문장:

■ if문(if부분, elif부분?, else부분?)





### 10. 먼저, 아래 코드를 잘 읽어 주세요: (13점)

```
def process():
    dap = 'Enter'

while True:
    # !!! 이동 - 출력
    move(0, 2)
    write('내 마음 속 키를 맞혀봐>')

choice = read_key()

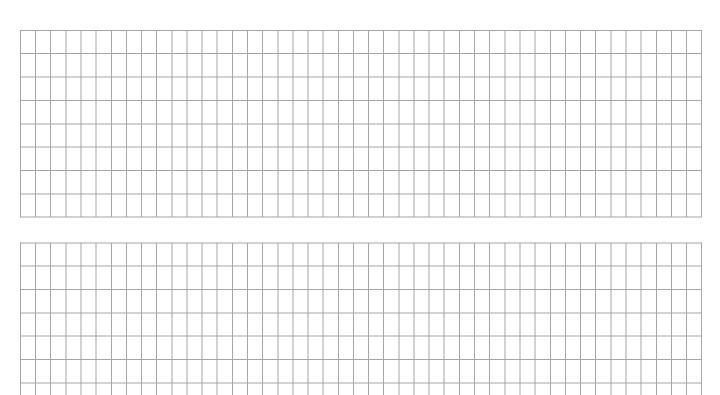
if choice == dap:
    # !!! 이동 - 출력
    move(0, 0)
    write('맞혔네.')

# !!! 이동 - 출력
    move(0, 1)
    write('이거 하나 맞혔다고 뿌듯해 하는건 아니겠지?')

break
```

위 코드에서 주석으로 표기된 부분들은 모두 '커서 이동'을 위한 move()와 '현재 커서 위치에 출력'을 위한 write()로 구성되어 있습니다. 매 번 이렇게 두 줄씩 적는 것은 번거로우므로, 이 두 작업을 한 번에 수행하는 새로운 함수를 만들어 사용하려고 합니다. 이러한 조건에 맞는 함수 write\_at()을 아래에 정의해 주세요.

주의: write\_at()은 두 작업을 수행하기 위한 적절한 인수를 가지고 있어야 합니다!



### 11. 아래 코드는 '3초 안에 Enter를 눌러야 이기는 게임' 코드의 일부입니다: (14점)

```
def do_the_game():
   dap = 'Enter'
   timeout = 3.0
   # time.clock()은 현재 시각에 해당하는 값(단위는 초)을 return하는 함수입니다.
   time_start = time.clock()
   while True:
      # process_input(dap)은,
      # 사용자가 키를 눌렀고 해당 키가 dap과 같으면 True를 return하는 함수입니다.
      # (사용자가 키를 안 눌렀거나, dap이 아닌 키를 누르면 False를 return합니다)
      if process_input(dap) == True:
         write('You won!')
         break
      #!!! 이 부분에 주목해 주세요!
      if check_time_over(time_start, timeout) == True:
         write('You lose...')
         break
```

주석으로 표시된 부분에서 사용되고 있는 함수 check\_time\_over()가 프로그램의 목표 및 주변 문맥에 맞게 잘 실행되도록 아래에 정의해 주세요.(부분점수 많아요)

진행 순서:

- 먼저, 게임의 이름과 각 변수들의 이름에 주목하면서 코드를 꼼꼼히 읽어 보세요.
- check\_time\_over()가 언제 이 함수가 True를 return해야 할 지 생각해 보세요.
- 위 예제에서 사용한 time.clock()을 활용하여 check\_time\_over()의 내용물을 채워 주세요.

