

1) 다음 중 프로그래밍 언어에 대한 설명으로 옳지 않은 것은?

1. 컴퓨터는 디지털 이진 숫자 0과 1로 구성되어 있다.
2. 프로그래밍 언어는 인간과 컴퓨터가 대화할 수 있도록 돕는 번역기 역할을 한다.
3. 소스 코드는 컴퓨터가 직접 이해할 수 있는 이진 코드로 작성된다.
4. C, C++, Java, Python 등 다양한 프로그래밍 언어가 존재한다.

이진 코드가 아니라 프로그래밍 언어 이다. 3번

2) 다음 중 컴퓨터의 기본 구조를 구성하는 장치에 포함되지 않는 것은 무엇인가?

1. 입력 장치
2. 제어 장치
3. 보조 기억 장치
4. 산술 논리 장치

보조 기억 장치(ex . ssd) 는 기본 구조 x 3번

3) 숫자 두 개를 입력받아 큰 수를 판별하는 코드입니다. 빈칸에 알맞은 답을 고르시오. 1번

```
A, B = map(int, input().strip())
if (          ):
    print(f'{A}가 {B}보다 큼니다. ')
else :
    print(f'{A}가 {B}보다 크지 않습니다. ')
```

1. A > B
2. A >=B
3. A < B
4. A <=B

4) 다음 보기 중 ? 에 대입하여 실행 불가능한 것을 모두 고르시오. 1, 2, 4

```
arg = ?
arg[0] = 0
print(arg)
```

1. (1, 2, 3)
2. {1, 2, 3}
3. [1, 2, 3] 수정 가능한 요소, 나머지는 수정이 안됨
4. 1, 2, 3 특히 애는 애초에 문법적으로 부적절

5) 다음코드에서 다음과 같이 출력되도록 for 문을 사용하여 solution을 완성하시오.

```
names = ['Alice', 'Bob', 'Charlie']
scores = [85, 92, 78]
majors = ['Math', 'Physics', 'Computer Science']

def solution(names, scores, majors):
    students = []
    for i in range(len(names)):
        student = {'name' : names[i], 'scores' : scores[i], 'majors' : majors[i]}
```

```
students.append(student)
return students
```

```
print(solution(names, scores, majors))
```

결과 :

```
[{'name': 'Alice', 'score': 85, 'major': 'Math'},
 {'name': 'Bob', 'score': 92, 'major': 'Physics'},
 {'name': 'Charlie', 'score': 78, 'major': 'Computer Science'}]
```

6) 다음 코드를 실행 시, num의 값이 변화하는 흐름으로 올바른 것은?

```
def fun1(num):
    num = fun3(num) + 1
    return num

def fun2(num):
    num = fun4(num) + 2
    return num

def fun3(num):
    num = fun2(num) + 3
    return num

def fun4(num):
    num += 4
    return num

def fun5(num):
    num = fun1(num) + 5
    return num

num = 5
fun5(num)
```

1.  $5 \rightarrow 6 \rightarrow 8 \rightarrow 11 \rightarrow 15 \rightarrow 20$
2.  $5 \rightarrow 9 \rightarrow 11 \rightarrow 14 \rightarrow 15 \rightarrow 20$
3.  $5 \rightarrow 10 \rightarrow 11 \rightarrow 14 \rightarrow 16 \rightarrow 20$
4.  $5 \rightarrow 9 \rightarrow 11 \rightarrow 12 \rightarrow 15 \rightarrow 20$

먼저 함수 쪽 타고 간다. 그래서 5 1 3 2 4 순서로 가서 더하면서 리턴 2번

7) 다음 코드는 할인된 가격을 계산하는 함수이다. 출력 결과가 아래와 같이 나오도록, 빈칸을 채우시오.

```
gum = 1000
def discount_gum(_____):
    global gum
    price = gum - (gum * num / 100)
    return price
```

```
print(discount_gum()) # 900.0
print(discount_gum(20)) # 800.0
```

디폴트에 대해 인지하고 있는지 **num = 10**

8) 다음 코드는 높은 수를 앞으로 보내는 선택 정렬 알고리즘이다. su 리스트가 완전히 정렬되려면 몇번 실행되어야 하는지 고르시오.

```
su = [5, 4, 7, 10, 6]

def swap(num_list):
    for i in range(len(num_list)-1):
        if num_list[i] < num_list[i+1]:
            num_list[i], num_list[i+1] = num_list[i+1], num_list[i]
    return num_list
```

1. 1번
2. 2번
3. 3번
4. 4번

**5 7 10 6 4 > 7 10 6 5 4 > 10 7 6 5 4**

**3번(어떻게 바뀌어야 더 좋을지 고민하다 버블정렬이 되어버림..)**

9) 다음 코드는 결제 금액을 계산해주는 클래스이다. 빈칸에 알맞게 채우시오.

```
market.py

class Product:
    def __init__(self, price, product = 100):
        self.price = price
        self.product = product
    def checkout(self, num):
        total = num * self.price
        self.product -= num
        return f'지불하실 금액 : {total} \n남은 수량 = {self.product}'
```

10) 9번의 market.py를 사용하여 main.py의 자식 클래스를 만들려고 한다. 빈칸에 알맞게 채우시오.

```
main.py
from market import Product

class Discount(Product):
    def __init__(self, price, percentage, product = 100):
        self.price = price
        self.percentage = percentage
        self.product = product
    def checkout(self, num):
        total = self.price * (1 - self.percentage/100) * num
        self.product -= num
        return f'지불하실 금액 : {total} \n남은 수량 = {self.product}'
```

**import와**

**self**

**오버라이딩**

11) 다음 tkinter 코드에서 빈칸을 채우시오.

```
import tkinter as tk

def show_selected():
    selected_value = selected_option.get()
    result_label.config(text=f"선택한 값: {selected_value}")

root = tk.Tk()
root.geometry("250x150")
options = ["사과", "바나나", "포도", "수박"]

selected_option = tk.StringVar()
selected_option.set(options[0])

option_menu = tk.OptionMenu(root, selected_option, *options)
option_menu.pack()

btn = tk.Button(root, text="선택한 값 출력", command=show_selected)
btn.pack()

result_label = tk.Label(root, text="선택한 값:")
result_label.pack()

root.mainloop()
```

12) 다음 코드의 빈 칸에 대해서 오류가 발생하는것을 고르시오.

```
file_write= open(path, _____, encoding='utf8')
file_write.write('안녕하세요')
```

1. 'r+'
2. 'w+'
3. 'a'
4. 'r'

4번은 읽기만 가능

13) 다음 코드의 빈칸을 알맞게 채우시오.

```
try:
    num = input("양수를 입력하세요: ")
    if num <= 0:
        _____ ZeroError("0 이하의 수는 입력할 수 없습니다. ")
        print(10 / num)
    except _____:
        print("0으로는 나눌 수 없습니다.")
    except _____:
        print("문자열을 입력받았습니다.")
except ZeroError as e:
    print(e)
```

raise와 예외 처리 항목에 대해 인지

raise / ZeroDivisionError / TypeError(ValueError도 가능)

14) 문자열을 입력받아 문자열 안의 모든 자연수의 합을 return 하는 함수를 만드시오.

```
import re
def solution(my_string):
    answer = 0
    p = re.compile('[1-9]')
    for i in range(len(my_string)):
        if p.match(my_string[i]):
            answer += int(my_string[i])
    return answer

return answer
```

15) 다음중 옳은 연결이 아닌것은?

1. 리스트 - [x for x in range(5)]
2. 세트 - {x for x in range(5)}
3. 딕셔너리 - {x : x\*\*2 in range(5)}
4. 튜플 - (x for x in range(5))

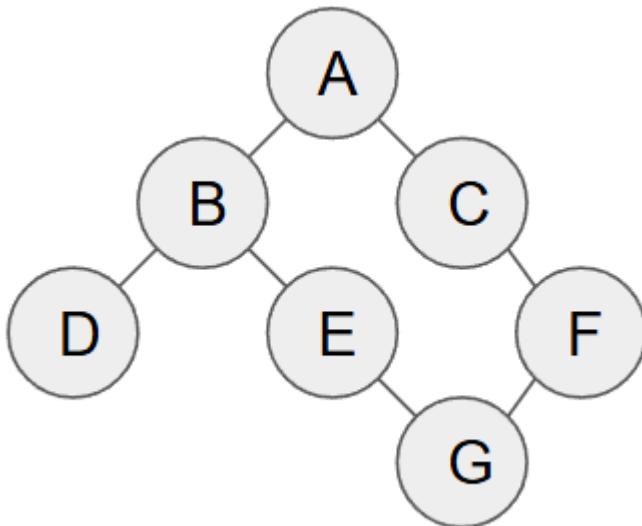
튜플 리프리헨션은 존재 x

16) 스택에 대한 설명으로 틀린것을 고르시오.

1. 데이터를 처리하는 기본 자료구조이다.
2. 선입선출(First In, First Out)의 형태로 데이터로 처리된다.
3. 괄호 검사와 같은 문자열 처리에서 사용된다.
4. 데이터의 삽입과 삭제가 상단(top)에서만 발생한다.

후입선출

17) 다음 그래프에서 시작 노드 A를 기준으로 깊이 우선 탐색(DFS) 과 너비 우선 탐색(BFS) 을 수행할 때, 각 알고리즘에 따른 노드 방문 순서를 각각 순서대로 나열하시오.



DFS : A > A B D E G F C

BFS : A > A B C D E F G

18) 다음 코드는 정n각형을 그리는 코드이다. 주어진 조건에 맞춰 빈 칸을 채우시오. (4칸)

선 색은 'black', 내부는 'plum'으로 채우기

속도는 5  
그리기 완료 후 모양 숨기기.

```
import turtle

num = int(input())
def polygon(num):
    t = turtle.Turtle()
    t.shape('turtle')
    t.speed(1)
    t.pensize(2)

    _____

    _____

    angle = 180 - (180 * (num-2)) / num
    for i in range(num):
        t.fd(100)
        t.left(angle)

    _____

    _____

    turtle.done()

polygon(num)
```

t.color('black', 'plum') / t.begin\_fill() / t.hideturtle() / t.end\_fill()

19) 섭씨에 1.8을 곱하고 32를 더하면 화씨라고 한다. 다음과 같은 섭씨의 리스트가 주어졌을 때 최댓값, 최솟값, 평균값을 화씨로 출력하는 함수를 numpy모듈을 사용하여 완성하여라.

(주의사항 - 소수점 4자리까지만 출력할것, numpy의 float형과 파이썬의 float형은 다른 타입)

```
import numpy as np
def cel_to_fah(cel_list):
    answer = []

    cel = np.array(cel_list)
    fah = cel * 1.8 + 32

    answer.append(float(np.max(fah)))
    answer.append(float(np.min(fah)))
    answer.append(round(float(np.mean(fah)), 4))

    return answer

num_list = [13.5, 15.2, 14.8, 16.0, 15.5, 17.1, 16.5]
```