알고리즘 – 주동현

2번

def gcd1(a, b):

    alist = []

    blist = []

    gcd\_list = []

    for i in range(1, a + 1):

        if a % i == 0:

            alist.append(i)

    for i in range(1, b + 1):

        if b % i == 0:

            blist.append(i)

    for i in alist:

        if i in blist:

            gcd\_list.append(i)

    print('60의 약수=', alist)

    print('28의 약수=', blist)

    return max(gcd\_list)

def gcd2(a, b):

    gcd = 1

    for i in range(1, min(a, b) + 1):

        if a % i == 0 and b % i == 0:

            gcd = i

    return gcd

def gcd3(a,b):

    while b!=0:

        r=a%b

        a=b

        b=r

    return a

print('60과 28의 최대공약수:', gcd1(60,28))

#print('60과 28의 최대공약수:', gcd2(60,28))

#print('60과 28의 최대공약수:', gcd3(60,28))

#2024/03/13 wed

3번

def gcd1(a, b):

    alist = []

    blist = []

    gcd\_list = []

    for i in range(1, a + 1):

        if a % i == 0:

            alist.append(i)

    for i in range(1, b + 1):

        if b % i == 0:

            blist.append(i)

    for i in alist:

        if i in blist:

            gcd\_list.append(i)

    print('60의 약수=', alist)

    #print('28의 약수=', blist)

    return max(gcd\_list)

def gcd2(a, b):

    gcd = 1

    for i in range(1, min(a, b) + 1):

        if a % i == 0 and b % i == 0:

            gcd = i

    return gcd

def gcd3(a,b):

    while b!=0:

        r=a%b

        a=b

        b=r

    return a

print('60과 28의 최대공약수:', gcd1(60,28))

#print('60과 28의 최대공약수:', gcd2(60,28))

#print('60과 28의 최대공약수:', gcd3(60,28))

#2024/03/13 wed

4번

def gcd1(a, b):

    alist = []

    blist = []

    gcd\_list = []

    for i in range(1, a + 1):

        if a % i == 0:

            alist.append(i)

            print(alist)

    for i in range(1, b + 1):

        if b % i == 0:

            blist.append(i)

    for i in alist:

        if i in blist:

            gcd\_list.append(i)

    print('60의 약수=', alist)

    #print('28의 약수=', blist)

    return max(gcd\_list)

def gcd2(a, b):

    gcd = 1

    for i in range(1, min(a, b) + 1):

        if a % i == 0 and b % i == 0:

            gcd = i

    return gcd

def gcd3(a,b):

    while b!=0:

        r=a%b

        a=b

        b=r

    return a

print('60과 28의 최대공약수:', gcd1(60,28))

#print('60과 28의 최대공약수:', gcd2(60,28))

#print('60과 28의 최대공약수:', gcd3(60,28))

#2024/03/13 wed

19번

import random as r

def max\_min(a):

    min = a[0]

    max = a[0]

    for i in range(0,len(a),1):

        if max > a[i]:

            max = a[i]

        if min < a[i]:

            min = a[i]

    print("가장 큰 수:", max)

    print("가장 작은 수:", min)

    return min, max

b = []

for i in range(0,10,1):

    b.append(r.randrange(1,10))

max\_min(b)

print(b)

20번

import random as r

def has\_same\_num(a, b):

    for num\_a in a:

        if num\_a in b:

            return True

    return False

def same\_num\_and\_print(a, b):

    same\_numbers = []

    for num\_a in a:

        if num\_a in b and num\_a not in same\_numbers:

            same\_numbers.append(num\_a)

    return same\_numbers

a = []

b = []

for i in range(10):

    a.append(r.randrange(1, 10))

    b.append(r.randrange(1, 10))

same\_numbers = same\_num\_and\_print(a, b)

if same\_numbers:

    print("같은숫자:", same\_numbers)

else:

    print("같은 숫자가 없습니다.")

print(a)

print(b)

print(has\_same\_num(a, b))

21번

# 공백 스택에서 pop을 시도해도 항목 수가 변하지 않으므로

# 3번의 공백 pop 시도 후의 항목 수는 그대로 0입니다.

# 그리고 7번의 pop을 하면 항목 수가 7개 줄어들게 됩니다.

# 따라서 총 25 - 7 = 18개의 항목이 남게 됩니다.

**25번**

**27번**

price = {'콩': 4500, '갈비탕': 9000, ' 돈가스': 8000}

print("메뉴이름: \t\t가격:")

for menu, cost in price.items():

    print("{:<10}\t\t{}".format(menu.strip(), cost))