

컴퓨터네트워크 학생 제출 문제풀이 보고서

제출자 정보 : ICT 융합공학부 / 정보통신소프트웨어 학과 / 201932030 / 송현교

목차

1. [3-01 / 전제민 / 201932052 / 날씨에 따른 준비물과 기온에 따른 권고문을 알려주는 프로그램](#)
2. [3-02 / 이태호 / 201932048 / 사용자에게 베팅할 금액을 입력 받아 맞추고 싶은 홀 짝을 선택하여 맞추는 게임 프로그램](#)
3. [3-03 / 박상수 / 201932025 / 사용자가 정수 2개, 사칙연산을 입력받고 선택한 연산에 대한 결과를 출력하는 프로그램](#)
4. [3-04 / 안명현 / 201932032 / PVP 자동 전투 시뮬레이션을 구현한 프로그램](#)
5. [3-05 / 나성진 / 202032017 / 떨어지는 사과를 바구니에 받아낼 때 필요한 이동 수 연산 프로그램](#)
6. [3-06 / 김채윤 / 201932017 / 칸이 3개인 슬롯머신 프로그램](#)
7. [3-07 / 이수현 / 201932043 / 파이썬으로 구현한 인디 포커](#)
8. [3-08 / 장혁 / 201932051 / 컴퓨터와 가위바위보 3선승](#)
9. [3-09 / 고병준 / 201832001 / 삼각형의 이등변, 정삼각형, 직각삼각형 여부 판별 프로그램](#)
10. [3-10 / 이지민 / 201932047 / 로또의 등수와 당첨여부 확인 프로그램](#)
11. [3-11 / 김현진 / 201932020 / 1부터 30까지의 정수 중 컴퓨터가 정한 숫자 맞추기](#)
12. [3-12 / 박예준 / 202032022 / 할인율에 따른 구매 금액 계산 프로그램](#)
13. [3-13 / 이동근 / 201932039 / 음료수 자판기 프로그램](#)
14. [3-14 / 최지훈 / 202032055 / 표준 몸무게와 비만도 계산법에 따른 비만도 측정 프로그램](#)
15. [3-15 / 서윤철 / 202032029 / 1~30 번호 중 랜덤 추첨 프로그램](#)
16. [3-16 / 김주현 / 201932013 / 파이썬으로 구현한 블랙 잭](#)
17. [3-17 / 용호중 / 201932033 / 유저와 컴퓨터의 단판 가위바위보 프로그램](#)
18. [3-18 / 이영민 / 201932044 / 로또 번호 당첨 여부 확인 프로그램](#)
19. [3-19 / 김동영 / 201932003 / 파이썬으로 구현한 틱택토](#)
20. [3-20 / 김주혁 / 201932012 / 키와 몸무게에 따른 정상체중과 bmi 수치 계산 프로그램](#)

21. [3-21 / 우민우 / 201932034 / 환율에 따른 환전 계산기 프로그램](#)
22. [3-22 / 이유진 / 201932045 / 소인수 분해 계산기 프로그램](#)
23. [3-23 / 김진용 / 201932014 / 20명의 남성을 키에 따라 운동종목 구분하는 프로그램](#)
24. [3-24 / 이세찬 / 201932041 / 파이썬으로 구현한 업 다운 게임](#)
25. [3-26 / 정성민 / 201932053 / 스택 자료구조 구현](#)
26. [3-27 / 김태균 / 201932012 / 공분산과 상관계수 계산 프로그램](#)
27. [3-28 / 석상훈 / 201932028 / 더 가까운 숫자를 입력할시 우승하는 프로그램](#)
28. [3-29 / 이성훈 / 201832062 / 주사위 게임](#)
29. [3-30 / 박현식 / 201932027 / 참참참 게임을 통한 농구관람료 벌이 시뮬레이션](#)
30. [3-31 / 김민수 / 201932006 / 업다운 게임](#)
31. [3-32 / 박상주 / 202121036 / 가위바위보 6판 대결 프로그램](#)
32. [3-33 / 최성현 / 202032054 / 학생의 이름과 점수에 따른 학점 조회 프로그램](#)
33. [4-01 / 전제민 / 201932052 / 랜덤 영어 or 한글 단어 타자 연습](#)
34. [4-02 / 이태호 / 201932048 / 사용자의 과목별 점수에 따른 성적 출력 프로그램](#)
35. [4-03 / 안명현 / 201932032 / rest api 구현 프로그램](#)
36. [4-04 / 이동근 / 201932039 / 딕셔너리를 이용한 최저 최고 기온 출력 프로그램](#)
37. [4-05 / 김채윤 / 201932017 / 비밀번호 오류 및 변경 시 나타나는 보안 코드 프로그램](#)
38. [4-06 / 장혁 / 201932051 / 전화번호부 추가, 삭제, 변경 및 조회 프로그램](#)
39. [4-07 / 박예준 / 202032022 / 영단어 퀴즈를 작성하는 프로그램](#)
40. [4-08 / 박상수 / 201932025 / 거스름돈 출력 프로그램](#)
41. [4-09 / 서윤철 / 202032029 / 상품명과 가격을 입력받아 딕셔너리에 저장하는 프로그램](#)
42. [4-10 / 김현진 / 201932020 / 집합이 같은 요소의 존재유무 판별 프로그램](#)
43. [4-11 / 김진용 / 201932014 / 성적에 따른 학점 출력 프로그램](#)
44. [4-12 / 김주현 / 201932013 / 당첨자 뽑기 프로그램](#)
45. [4-13 / 나성진 / 리스트를 이용한 스택 구현 프로그램](#)

46. [4-14 / 이유진 / 201932045 / 2인이 함께 하는 전화번호 맞추기 프로그램](#)
47. [4-15 / 이수현 / 201932043 / 입력된 임의의 값 맞추기 프로그램](#)
48. [4-16 / 김동영 / 201932003 / 파이썬으로 구현한 끝말잇기 프로그램](#)
49. [4-18 / 김태균 / 202032012 / 한쪽 끝이 막힌 주차장 프로그램 구현](#)
50. [4-19 / 고병준 / 201832001 / 궁합이 좋은 음식 추천 및 브랜드 출력](#)
51. [4-20 / 이지민 / 201932047 / 평균학점 3.5를 얻기 위한 추가 최소학점 출력](#)
52. [4-21 / 이세찬 / 201932041 / 딕셔너리를 리스트로 변환하는 프로그램](#)
53. [4-22 / 최성현 / 202032054 / 판매중인 과일 조회 및 추가, 삭제 프로그램](#)
54. [4-23 / 우민우 / 201932034 / 로또 당첨 여부 확인 프로그램](#)
55. [4-24 / 용호중 / 201932033 / 통장에서 입금 및 출금 시뮬레이션](#)
56. [4-25 / 최지훈 / 202032055 / 미국에 있는 각각의 종의 나무 비율 확인 프로그램](#)
57. [4-26 / 석상훈 / 201932028 / 100점 만점의 점수에 대해 각 등급의 개수 카운트 프로그램](#)
58. [4-27 / 김민규 / 201932005 / 점심 메뉴 랜덤 출력 프로그램](#)
59. [4-28 / 정성민 / 201932053 / 한달 용돈에 대한 소비습관 계산 프로그램](#)
60. [4-29 / 박주혁 / 202032025 / 무작위 한자리수 맞추기 프로그램](#)
61. [4-30 / 이영민 / 201932044 / 아이스크림 자판기 프로그램](#)
62. [4-31 / 김주혁 / 201932012 / 카페 키오스크 프로그램 구현](#)
63. [4-32 / 김민수 / 201932006 / 점심, 저녁 메뉴 추천 프로그램](#)
64. [4-33 / 박현식 / 농구 포지션 추천 프로그램](#)
65. [5-01 / 전제민 / 201932052 / 함수를 이용한 계산기 프로그램](#)
66. [5-02 / 이태호 / 201932048 / 3과목의 점수에 대한 평균, 등급 계산 프로그램](#)
67. [5-03 / 이동근 / 201932039 / 로또 번호를 랜덤으로 생성하는 프로그램](#)
68. [5-05 / 장혁 / 201932051 / bmi 계산 프로그램](#)
69. [5-06 / 김현진 / 함수를 이용한 계산기 프로그램](#)
70. [5-07 / 박상수 / 201932025 / 월별 계절 출력 프로그램](#)

71. [5-08 / 박예준 / 202032022 / 주사위 2개를 던져 나온 수에 따른 승패처리 시뮬레이션](#)
72. [5-09 / 김주현 / 201932013 / 도시의 이름을 영어로 입력받고 출력하는 프로그램](#)
73. [5-10 / 이유진 / 201932045 / 지뢰 찾기 구현](#)
74. [5-11 / 이지민 / 201932047 / 투자에 따른 순이익, 가치 변동율 측정 프로그램](#)
75. [5-12 / 이수현 / 201932043 / 계산기 제작 프로그램](#)
76. [5-13 / 김태균 / 202032012 / 사용자가 입력한 값의 합, 최댓값, 최솟값, 평균, 분산 계산](#)
77. [5-14 / 김채윤 / 201932017 / 게임 플레이어 인터페이스 구현](#)
78. [5-15 / 안명현 / 파이썬 내장 함수 sorted\(\) 직접 구현](#)
79. [5-16 / 고병준 / 201832001 / 영화와 뮤지컬 선택 후 나이에 따른 가격 출력](#)
80. [5-17 / 최지훈 / 202032055 / 파이썬을 이용해 구현한 근의 공식](#)
81. [5-18 / 김동영 / 201932003 / 그림 그리기 프로그램](#)
82. [5-19 / 이영민 / 201932044 / 시험성적에 따른 등급 조회 프로그램](#)
83. [5-20 / 김민규 / 201932005 / 기초 대사량 구하기 프로그램](#)
84. [5-21 / 이세찬 / 201932041 / 로또 번호 출력 프로그램](#)
85. [5-22 / 용호중 / 201932033 / 일일 카페인 권장량 출력 프로그램](#)
86. [5-23 / 우민우 / 201932024 / 회원가입 및 로그인 구현](#)
87. [5-24 / 나성진 / 202032017 / N장의 카드를 순서에 따라 재배열 및 제거하는 프로그램](#)
88. [5-25 / 김진용 / 201932014 / 메뉴 고르기 프로그램](#)
89. [5-26 / 정성민 / 201932053 / 랜덤으로 받은 수를 내림차순으로 직접 정렬하는 프로그램](#)
90. [5-27 / 박현식 / 201932027 / 키, 체중에 따른 bmi 지수 계산 프로그램](#)
91. [5-28 / 김민수 / 201932006 / 구구단 출력 프로그램](#)
92. [5-29 / 김주혁 / 201932012 / 리모콘으로 조절하는 채널과 볼륨 프로그램](#)
93. [5-30 / 석상훈 / 201932028 / 세 수 중 가장 큰 수, 가장 작은 수를 구하는 계산기](#)

3-01 / 전제민 / 201932052

1) 문제 설명

문제는 날씨에 따른 준비물과 기온에 따른 권고문을 알려주는 프로그램입니다.

2) 문제 조건

- ① 조건문 if 문을 사용해야 하며 그중 elif 구문을 반드시 사용할 것.
- ② 날씨는 비, 눈, 미세먼지, 강풍 4 개로 하고 준비물 자유
- ③ 기온은 상식적인 범위 및 권고문 자유롭게 작성

```
weather = input("오늘의 날씨를 입력하세요. ( 비 / 눈 / 미세먼지 / 강풍 ) >>> ")

if(weather == "눈"):
    print("우산을 챙기세요. 길이 미끄러우니 조심하세요.")
elif(weather == "비"):
    print("우산을 챙기세요. 신발이 젖을 수 있으니 조심하세요.")
elif(weather == "미세먼지"):
    print("마스크를 착용하십시오.")
elif(weather == "강풍"):
    print("외투를 챙기십시오.")

temp = int(input("기온을 입력하십시오. "))
if 30 <= temp :
    print("너무 덥습니다.")
elif 10 <= temp:
    print("괜찮은 날씨입니다.")
elif 0 <= temp:
    print("쌀쌀한 날씨입니다..")
else:
    print("매우 추운 날씨입니다.")
```

3-02 / 이태호 / 201932048

1) 문제 설명

사용자에게 베팅할 금액을 입력 받아 맞추고 싶은 홀 짝을 선택해 맞추는 게임입니다.

2) 문제 조건

- ① 매판마다 사용자가 계속할 수 있거나 그만 할 수 있게 설정할 것
- ② 배당은 자유롭게 설정 가능하되 베팅 금액이 0 이 될 시 자동종료
- ③ 난수 생성과 if 문을 활용하려 코딩할 것

```
from random import randint
print("홀짝 게임입니다. 당신의 초기자금은 1000 원이며, 승리시 배율은 2 배입니다.")
moneyA = 1000
while(True):
    coin = randint(1,50)
    while(True):
        money = int(input("베팅할 금액을 입력하십시오. >>> "))
        if(money > moneyA):
            print("보유금액보다 큰 금액을 베팅할 수 없습니다. 다시 베팅하십시오.")
            continue
        else:
            break
    moneyA = moneyA - money
    num = input("홀 / 짝 >>> ")

    if(num == "홀"):
        if(coin % 2 == 1):
            moneyA = moneyA + (money * 3)
            print("승리하셨습니다 ! 현재 보유금액은 ", moneyA, "원 입니다.")

        else:
            print("패배하셨습니다.. 현재 보유금액은 ", moneyA, "원 입니다.")
        if(moneyA <=0):
            print("보유금액이 0 원이 되었습니다. 게임을 종료합니다.")
            quit()
        if(answer == "Y"):
            continue
        elif(answer == "N"):
            quit()
    elif(num == "짝"):
        if(coin % 2 == 0):
            moneyA = moneyA + (money * 3)
            print("승리하셨습니다 ! 현재 보유금액은 ", moneyA, "원 입니다.")

        else:
            print("패배하셨습니다.. 현재 보유금액은 ", moneyA, "원 입니다.")
        if(moneyA <=0):
            print("보유금액이 0 원이 되었습니다. 게임을 종료합니다.")
            quit()
    answer = input("한 번 더 해보시겠습니까 ? (Y / N) >>> ")
    if(answer == "Y"):
        continue
    elif(answer == "N"):
```

```
quit()
if(moneyA <=0):
    print("보유금액이 0원이 되었습니다. 게임을 종료합니다.")
    quit()
```


3-03 / 박상수 / 201932025

1) 문제 설명

사용자가 정수 2 개, 사칙연산을 입력 받고 선택한 연산에 대한 결과를 출력하는 프로그램입니다.

2) 문제 조건

- ① 사용자가 정수 2 개와 사칙연산을 입력하도록 설정하기
- ② 나누기시 첫번째 정수가 0 일때 나누기가 안되도록 설정하기
- ③ 연산선택시 선택한 연산으로 가도록 if 문을 활용하여 코딩하기
- ④ 종료를 선택하지 않는 이상 계속 계산할 수 있도록 할 것

```
while(True):
    sign=input("수행할 연산을 입력하십시오. ( + - * / 종료 )")
    if (sign == "종료"):
        break
    n = int(input('첫번째 정수를 입력하십시오. >>>'))
    m = int(input('두번째 정수를 입력하십시오. >>>'))
    if(sign == "+"):
        print(n, " + ", m, " = ", n+m)
    elif(sign == "-"):
        print(n, " - ", m, " = ", n-m)
    elif(sign == "*"):
        print(n, " * ", m, " = ", n*m)
    elif(sign == "/"):
        if((n == 0) or (m == 0)):
            print('Devision By Zero')
        else:
            print(n, " / ", m, " = ", n/m)
```

3-04 / 안명현 / 201932032

1) 문제 설명

PVP 자동 전투 시뮬레이션

플레이어의 HP와 방어력, 공격력, 치명타 확률, 명중률, 회피율을 입력받고

한 플레이어가 사망할 때까지 한 턴씩 번갈아 가며 공격하는 시뮬레이션을 하는 프로그램을 구현하시오.

2) 문제 조건

- ① 데미지 공식 : 공격 플레이어 공격력 - 방어 플레이어 방어력 (단, 0보다 낮을 경우 1으로 보정 / 무한 루프 방지)
- ② 치명타 발생시 : 데미지를 2배로 처리
- ③ 명중률 공식 : 공격 플레이어 명중률 - 방어 플레이어 회피율 (단, 100이 넘어갈 시 100으로 보정)

```
from random import randint
# 0:HP, 1:방어력, 2:공격력, 3:치명타 확률, 4:명중률, 5:회피율
while(True):
    player1 = {'HP':0, '방어력':0, '공격력':0, '치명타 확률':0, '명중률':0, '회피율':0}
    print("Player 1의 능력치를 설정하겠습니다.")
    for i in range(0,6):
        temp = str(list(player1.keys())[i])
        print(list(player1.keys())[i],end = '')
        player1[temp] = int(input("를 입력하시오 >>> "))
    print(player1)
    answer = input("Player 1의 능력치를 이대로 설정하시겠습니까 ? ( Y / N ) ")
    if(answer == "Y"):
        break
    elif(answer == "N"):
        print("능력치를 재설정합니다.")
        continue

while(True):
    player2 = {'HP':0, '방어력':0, '공격력':0, '치명타 확률':0, '명중률':0, '회피율':0}
    print("Player 2의 능력치를 설정하겠습니다.")
    for i in range(0,6):
        temp = str(list(player2.keys())[i])
        print(list(player2.keys())[i],end = '')
        player2[temp] = int(input("를 입력하시오 >>> "))
    print(player2)
    answer = input("Player 2의 능력치를 이대로 설정하시겠습니까 ? ( Y / N ) ")
    if(answer == "Y"):
        break
    elif(answer == "N"):
        print("Player 2 능력치를 재설정합니다.")
        continue

def damage(damage, defense):
    if(damage > defense):
        return damage - defense
    elif(damage < defense):
        return 1
```

```

    elif(damage == defense):
        return 1

def critical(critical):
    rdm = int(100 / critical)
    crit = randint(1,rdm)
    if(crit == rdm):
        print("크리티컬 !!!")
        return True
    else:
        return False

def accuracy(accuracy,evasion):
    if((accuracy < evasion) or (accuracy - evasion == 0)):
        a1 = 99
        acr = randint(1,a1)
    else:
        a1 = int(100 / (accuracy - evasion))
        acr = randint(1,a1)
    if(acr == a1):
        return True
    else:
        return False

print("전투를 시작합니다.")
print("Player 1의 선폭.")
while(True):
    print("Player 1의 공격 1")
    if(accuracy(player1['명중률'],player2['회피율']) == True):
        if(critical(int(player1['치명타 확률'])) == True):
            print("크리티컬로 ", end='')
            print("공격이 명중하였습니다 ! 가한 데미지 = ",((damage((int(player1['공격력'])),(int(player2['방어력'])))) * 2)
            player2['HP'] = player2['HP'] -
            (damage((int(player1['공격력'])),(int(player2['방어력'])) * 2)
            print("Player 1의 현재 HP = ", int(player1['HP']), "Player 2의 현재 HP = ",
            int(player2['HP']))
        else:
            print("공격이 명중하였습니다 ! 가한 데미지 = ",damage((int(player1['공격력'])),(int(player2['방어력'])))
            player2['HP'] = player2['HP'] -
            damage((int(player1['공격력'])),(int(player2['방어력'])))
            print("Player 1의 현재 HP = ", int(player1['HP']), "Player 2의 현재 HP = ",
            int(player2['HP']))
        else:
            print("공격이 빔나갔습니다.")
            print("Player 1의 현재 HP = ", int(player1['HP']), "Player 2의 현재 HP = ",
            int(player2['HP']))

    ###
    print("Player 2의 공격 !")
    if(accuracy((player2['명중률']),(player1['회피율'])) == True):
        if(critical == True):
            if(critical(player2['치명타 확률'] == True)):
                print("크리티컬로 ", end = '')

```

```

        print("공격이 명중하였습니다 ! 가한 데미지 =
",((damage((int(player2['공격력']),int(player1['방어력'])))) * 2)
        player1['HP'] = player1['HP'] -
((damage(int(player2['공격력']),int(player1['방어력'])) * 2)
        print("Player 1의 현재 HP = ", int(player1['HP']), "Player 2의 현재 HP = ",
int(player2['HP']))
    else:
        print("공격이 명중하였습니다 ! 가한 데미지 =
", (damage((int(player2['공격력']),int(player1['방어력']))))
        player1['HP'] = player1['HP'] - (damage(int(player2['공격력']),int(player1['방어력'])))
        print("Player 1의 현재 HP = ", int(player1['HP']), "Player 2의 현재 HP = ",
int(player2['HP']))
    else:
        print("공격이 빗나갔습니다.")
        print("Player 1의 현재 HP = ", int(player1['HP']), "Player 2의 현재 HP = ",
int(player2['HP']))

#####
if(player1['HP'] <= 0):
    print("Player 1의 패배입니다.")
    break
elif(player2['HP'] <= 0):
    print("Player 2의 패배입니다.")
    break

```

3-05 / 나성진 / 202032017

1) 문제 설명

상근이는 오락실에서 바구니를 옮기는 오래된 게임을 한다.

스크린은 N 칸으로 나누어져 있다. 스크린의 아래쪽에는 M 칸을 차지하는 바구니가 있다. ($M < N$)

플레이어는 게임을 하는 중에 바구니를 왼쪽이나 오른쪽으로 이동할 수 있다.

하지만, 바구니는 스크린의 경계를 넘어가면 안 된다.

가장 처음에 바구니는 왼쪽 M 칸을 차지하고 있다.

스크린의 위에서 사과 여러 개가 떨어진다. 각 사과는 N 칸중 한 칸의 상단에서 떨어지기 시작하며, 스크린의 바닥에 닿을 때까지 직선으로 떨어진다. 한 사과가 바닥에 닿는 즉시, 다른 사과가 떨어지기 시작한다.

바구니가 사과가 떨어지는 칸을 차지하고 있다면, 바구니는 그 사과가 바닥에 닿을 때, 사과를 담을 수 있다. 상근이는 사과를 모두 담으려고 한다.

이때, 바구니의 이동 거리의 최솟값을 구하는 프로그램을 작성하시오.

2) 문제 조건

-입력-

첫째 줄에 N 과 M 이 주어진다. ($1 \leq M < N \leq 10$)

둘째 줄에 떨어지는 사과의 개수 J 가 주어진다. ($1 \leq J \leq 20$)

다음 J 개 줄에는 사과가 떨어지는 위치가 순서대로 주어진다.

-출력-

모든 사과를 담기 위해 바구니가 이동해야 하는 거리의 최솟값을 출력한다.

```
space, back = map(int, input("스크린의 길이와 바구니의 길이를 입력하세요: ").split())
num = int(input("떨어뜨릴 사과의 갯수를 입력하세요: "))
front = 1
move_cnt = 0
print("사과를 n 번 떨어뜨리세요")
for i in range(num):
    location = int(input())
    if location < front:
        move_cnt += front - location
        back -= front - location
        front = location
        print("front:", front, "back:", back, "move_cnt", move_cnt)
    elif back < location:
        move_cnt += location - back
        front += location - back
        back = location
        print("front:", front, "back:", back, "move_cnt", move_cnt)
    else:
        print("front:", front, "back:", back, "move_cnt", move_cnt)
print("사과를 모두 담기 위해 움직여야 하는 최소거리는 ", move_cnt, "칸입니다", sep='')
```

3-06 / 김채운 / 201932017

1) 문제 설명

칸이 3 개인 슬롯머신을 구현하시오. 단 숫자마다 등장할 확률이 다르며, count 가 100 되면 jackpot 이 나오게 하시오

2) 문제 조건

- ① while 문을 사용하시오
- ② 난수 생성을 하시오

```
from random import randint
money = 1000
count = 0

while(True):
    count += 1
    if(count == 100):
        print("Jackpot !!! [ 7 7 7 ] 게임을 종료합니다.")
        break

    print("현재 당신의 보유금은 ", int(money), "원 입니다.")
    while(True):
        coin = int(input("투입할 금액을 입력하십시오.>>> "))
        if(coin > money):
            print("보유금보다 큰 금액을 투입할 수 없습니다. 다시 입력하십시오.")
        else:
            break

    first = randint(0,2)
    second = randint(0,2)
    third = randint(0,2)
    category = ["사과", "바나나", "7"]

    if(category[first] == "사과" and category[second] == "사과" and category[third] == "사과"):
        print("사과 ", "사과 ", "사과 ")
        print("당첨, 배율 : 3 배")
        money = money + (coin * 3)
        continue

    elif(category[first] == "사과" and category[second] == "사과" and category[third] != "사과"):
        print("사과 ", "사과 ", category[third])
        print("당첨, 배율 : 1.5 배")
        money = money + (coin * 1.5)
        continue

    elif(category[first] == "사과" and category[second] != "사과" and category[third] == "사과"):
        print("사과 ", category[second] , "사과 ")
        print("당첨, 배율 : 1.5 배")
        money = money + (coin * 1.5)
        continue
```

```

elif(category[first] != "사과" and category[second] == "사과" and category[third] == "사과"):
    print(category[first], " 사과 ", " 사과 ")
    print("당첨, 배율 : 1.5 배")
    money = money + (coin * 1.5)
    continue

#바나나

if(category[first] == "바나나" and category[second] == "바나나" and category[third] ==
"바나나"):
    print(" 바나나 ", " 바나나 ", " 바나나 ")
    print("당첨, 배율 : 4 배")
    money = money + (coin * 4)
    continue

elif(category[first] == "바나나" and category[second] == "바나나" and category[third] !=
"바나나"):
    print(" 바나나 ", " 바나나 ", category[third])
    print("당첨, 배율 : 1.6 배")
    money = money + (coin * 1.6)
    continue

elif(category[first] == "바나나" and category[second] != "바나나" and category[third] ==
"바나나"):
    print(" 바나나 ", category[second] , " 바나나 ")
    print("당첨, 배율 : 1.6 배")
    money = money + (coin * 1.6)
    continue

elif(category[first] != "바나나" and category[second] == "바나나" and category[third] ==
"바나나"):
    print(category[first], " 바나나 ", " 바나나 ")
    print("당첨, 배율 : 1.6 배")
    money = money + (coin * 1.6)
    continue

#7

if(category[first] == "7" and category[second] == "7" and category[third] == "7"):
    print(" 7 ", " 7 ", " 7 ")
    print("Jackpot !!! , 배율 : 10 배")
    money = money + (coin * 10)
    continue

elif(category[first] == "7" and category[second] == "7" and category[third] != "7"):
    print(" 7 ", " 7 ", category[third])
    print("당첨, 배율 : 2 배")
    money = money + (coin * 2)
    continue

elif(category[first] == "7" and category[second] != "7" and category[third] == "7"):
    print(" 7 ", category[second] , " 7 ")
    print("당첨, 배율 : 2 배")
    money = money + (coin * 2)

```

```
continue
```

```
elif(category[first] != "7" and category[second] == "7" and category[third] == "7"):
```

```
    print(category[first], " 7 ", " 7 ")
```

```
    print("당첨, 배율 : 2 배")
```

```
    money = money + (coin * 2)
```

```
    continue
```

```
else:
```

```
    print(category[first], "", category[second], "", category[third])
```

```
    print("실패")
```

```
    money = money - coin
```


3-07 / 이수현 / 201932043

1) 문제 설명

파이썬으로 인디언 포커를 만들려 한다. 단, 돈을 거는 데에는 제한이 없음.

또한 AI와의 대결을 만들어서 하려고 한다.

2) 문제 조건

① 플레이어와 AI에게 1부터 10까지 들어있는 배열을 각각 부여한다.

② 플레이어와 AI가 부여받은 배열을 랜덤으로 재배치해야한다.

③ 플레이어와 AI의 리스트가 다 되는 경우 또는 AI, 플레이어 둘 중 한명의 돈이 다 없어지는 경우 게임을 종료시켜 결과를 보여주도록 한다.

④ while 문을 이용하시오

⑤ 플레이어와 AI가 게임을 시작할때 돈이 5원이 차감되고 플레이어가 이번 게임을 포기하는 경우 ai는 다시 5원을

획득하도록 하시오.

```
print("인디언 포커 게임입니다.")

PLAYER = ['A','B','C','D','E','F','G','H','I','J']
AI = ['A','B','C','D','E','F','G','H','I','J']
pmoney = 30
aimoney = 30

random.shuffle(PLAYER)
random.shuffle(AI)

print("양 측의 카드를 무작위로 재배치하였습니다. 알파벳이 큰 쪽이 승리합니다.")

for i in range(10):
    if(pmoney <= 0):
        print("Player의 패배입니다. AI의 보유금 : ",aimoney)
        break
    if(aimoney <= 0):
        print("AI의 패배입니다. Player의 보유금 : ",pmoney)

    answer = input("( 시작 / 포기 )")
    if(answer == "시작"):
        print("게임 시작 !")
        pmoney = pmoney - 5
        aimoney = aimoney - 5

    elif(answer == "포기"):
        break
        aimoney = aimoney + 5

print("[ Player : ", PLAYER[i], " ] ", "[ AI : ", AI[i], " ]")
if(PLAYER[i] > AI[i]):
    print("Player의 승리 ! 5원을 획득합니다.")
    pmoney = pmoney + 5
```

```
    aimoney = aimoney - 5

elif(PLAYER[i] == AI[i]):
    print("동점 !")

else:
    print("AI 의 승리 ! 5 원을 잃습니다.")
    pmoney = pmoney - 5
    aimoney = aimoney + 5
```

3-08 / 장혁 / 201932051

1) 문제 설명

컴퓨터와 가위바위보를 해서 3 번 이기면 끝나는 프로그램을 만들었습니다.

2) 문제 조건

- ① 내가 낼 수 있는 것은 가위, 바위, 보 이렇게 3 가지
- ② 컴퓨터에게는 패배하거나 비길 수 있음.
- ③ if, else 문을 이용하라
- ④ 3 승이 되면 프로그램을 종료

```
from random import randint
count = 0

while(True):
    ai = randint(0,2)
    if(ai == 0):
        ai = "가위"
    elif(ai == 1):
        ai = "바위"
    elif(ai == 2):
        ai = "보"

    player = input("( 가위 / 바위 / 보 )")

    if(player == "가위"):
        if(ai == "가위"):
            print("무승부 !")
        elif(ai == "주먹"):
            print("AI 승리 !")
        elif(ai == "보"):
            print("Player 승리 !")
            count += 1

    elif(player == "바위"):
        if(ai == "바위"):
            print("무승부 !")
        elif(ai == "보"):
            print("AI 승리 !")
        elif(ai == "가위"):
            print("Player 승리 !")
            count += 1

    elif(player == "보"):
        if(ai == "보"):
            print("무승부 !")
        elif(ai == "가위"):
            print("AI 승리 !")
        elif(ai == "주먹"):
```

```
        print("Player 승리 !")
        count += 1

if(count >= 3):
    print("Player 의 우승입니다 . 게임을 종료합니다 .")
    quit()
```

3-09/ 고병준 / 201832001

1) 문제 설명

삼각형의 세 변의 길이를 입력하여 삼각형의 종류 (ex. 정삼각형, 이등변 삼각형...) 등을 판별하는 프로그램입니다.

2) 문제 조건

- ① 세 수는 모두 정수이다.
- ② 조건문을 사용하여 작성하라
- ③ 삼각형, 정삼각형, 직각삼각형, 이등변 삼각형, 삼각형이 아닐 경우를 판별하라

```
a, b, c = input('삼각형의 세 변을 입력하시오.').split()
a = int(a)
b = int(b)
c = int(c)

if a>b and a>c:
    num1 = b
    num2 = c
    num3 = a
elif b>a and b>c:
    num1 = a
    num2 = c
    num3 = b
else:
    num1 = a
    num2 = b
    num3 = c

if num1+num2 <= num3:
    print(" 삼각형이 아닙니다. ")
elif num1==num2 and num2==num3:
    print(" 정삼각형 입니다. ")
elif num1==num2 or num1==num3 or num2 == num3:
    print(" 이등변 삼각형 입니다. ")
elif num1**2+num2**2 == num3**2:
    print(" 직각 삼각형 입니다. ")
else:
    print(" 삼각형 입니다. ")
```

3-10/ 이지민 / 201932047

1) 문제 설명

랜덤 로또번호를 생성하고 내가 입력한 번호와 같은지와, 등수 및 당첨여부를 확인시켜주는 프로그램을 작성하시오.

(로또번호는 1~45 까지)

2) 문제 조건

- ① random 함수를 사용하시오.
- ② 배열을 사용하시오.
- ③ if else 문을 사용하시오.

```
from random import randint
lotto = []
while(True):
    num = randint(1,45)
    if(num not in lotto):
        lotto.append(num)
    if(len(lotto) == 6):
        break

inputStr = input("로또 번호를 6 개 입력하시오. >>> ").split(" ")
print("당첨 번호 = ", lotto)
my = []
for i in inputStr:
    my.append(int(i))

same = 0
for j in my:
    if j in lotto:
        same += 1

score = [0,6,5,4,3,2,1]
if same == 0:
    print("다음 기회를 노려보세요.")
else:
    print(same, "개 맞았습니다 !", score[same], "등 당첨!")
```

3-11/ 김현진 / 201932020

1) 문제 설명

1 부터 30 까지의 정수 중 컴퓨터가 랜덤으로 정한 숫자를 알아맞히는 랜덤 숫자 게임을 만들었습니다.

2) 문제 조건

- ① 정답이 아닐 시 사용자가 입력한 숫자보다 높은지(UP), 낮은지(DOWN) 알려주는 힌트를 출력해주세요.
- ② 게임이 종료되면 몇번만에 정답을 맞췄는지도 출력해보세요.
- ③ if,else,random 을 이용하라.

```
from random import randint

AI = randint(1,30)
count = 1

while(True):
    answer = int(input("1 부터 30 까지 랜덤한 숫자가 입력되었습니다. 맞춰 보세요. >>> "))

    if(answer == AI):
        print("정답입니다 ! ", count, "번 만에 정답을 맞추셨습니다.")
        break
    else:
        if(answer < AI):
            print("오답입니다. 더 큰 수를 입력하셔야 합니다.")
            count += 1
        elif(answer > AI):
            print("오답입니다. 더 작은 수를 입력하셔야 합니다.")
            count += 1
```

3-12/ 박예준 / 202032022

1) 문제 설명

등급에 따른 할인율을 적용하여 구매한 금액을 계산하는 프로그램입니다.

2) 문제 조건

- ① 등급에 따른 할인율 지정 (할인율 A 등급-30%, B 등급-15%, C 등급-8% D 등급-5% E 등급-1%).
- ② m-class 변수를 만들어 등급을 A,B,C,D,E 중 하나를 선택하여 저장
- ③ if~elif~else 을 이용 등급별 할인율이 적용된 sale_price 금액을 계산

```
m_class = "A"
price = 10000

sale = 0

if(m_class == "A"):
    print("A 등급의 할인율은 30%입니다.")
    sale = price * 0.3
elif(m_class == "B"):
    print("B 등급의 할인율은 15%입니다.")
    sale = price * 0.15
elif(m_class == "C"):
    print("C 등급의 할인율은 8%입니다.")
    sale = price * 0.08
elif(m_class == "D"):
    print("D 등급의 할인율은 5%입니다.")
    sale = price * 0.05
elif(m_class == "E"):
    print("E 등급의 할인율은 1%입니다.")
    sale = price * 0.01

print("결제해야 할 금액은 ",price - sale, "입니다.")
```


3-13/ 이동근 / 201932039

1) 문제 설명

음료수 자판기를 구현했습니다. 돈을 넣으면 잔액을 알려주고 금액이 남으면 음료수를 계속 뽑아 마실 수 있습니다.

2) 문제 조건

- ① 0~4 버튼 이외의 숫자를 선택하면 다시 선택하도록 프로그램을 구현하시오
- ② 0 버튼을 선택하면 프로그램이 종료되도록 구현하시오
- ③ 잔액이 0 이 되면 프로그램이 종료되도록 구현하시오
- ④ 잔액이 부족하면 돈을 더 넣고 음료수를 뽑을 수 있도록 구현하시오
- ⑤ 잔액을 표시되게 구현하시오

```
money = int(input("넣을 금액은 얼마입니까 ? >>> "))

while(True):
    while(True):
        button = int(input("0.프로그램 종료 / 1.생수(900\ ) / 2.콜라(1,500\ ) / 3.사이다(1,300\)"))
        if(button < 0 or button > 3):
            print("다시 시도하십시오.")
            continue
        else:
            break
    if(button == 0):
        quit()
    elif(button == 1):
        if(money < 900):
            money = money + int(input("잔액이 부족합니다. 잔액을 얼마 더 넣으시겠습니까 ?"))
            print("잔액이 충전되었습니다. 현재 충전되어 있는 금액 : ",money)
            continue
        else:
            money = money - 900
            print("감사합니다.")
    elif(button == 2):
        if(money < 1500):
            money = money + int(input("잔액이 부족합니다. 잔액을 얼마 더 넣으시겠습니까 ?"))
            print("잔액이 충전되었습니다. 현재 충전되어 있는 금액 : ",money)
            continue
        else:
            money = money - 1500
            print("감사합니다.")
    elif(button == 3):
        if(money < 1300):
            money = money + int(input("잔액이 부족합니다. 잔액을 얼마 더 넣으시겠습니까 ?"))
            print("잔액이 충전되었습니다. 현재 충전되어 있는 금액 : ", money)
            continue
        else:
            money = money - 1300
            print("감사합니다.")
```

3-14/ 최지훈 / 202032055

1) 문제 설명

키와 몸무게를 이용하여 비만도를 측정할 수 있다.

아래와 같이 표준 몸무게와 비만도 계산법이 있을 때, 비만도를 측정하는 프로그램을 작성하시오.

4 * 표준몸무게

키에 따른 표준몸무게	공식
키가 150 미만일 때	(실제 키 - 100)
키가 150 이상 160 미만일 때	(실제 키 - 150) / 2 + 50
키가 160 이상일 때	(실제 키 - 100) * 0.9

5 * 비만도 계산 공식

$$6 \quad \text{비만도} = (\text{실제 몸무게} - \text{표준몸무게}) * 100 / \text{표준 몸무게}$$

7 * 비만도에 따른 등급 판정

등급	비만도 수치
10 이하	정상
10~20 이하	과체중
20 초과	비만

ex) 키가 150Cm 이고, 몸무게는 60kg 이라고 하자.

$$\text{표준 몸무게} = (150 - 150) / 2 + 50 = 50 \text{ kg}$$

비만도 = $(60 - 50) * 100 / 50 = 20$

따라서 비만도가 20 이므로 "과체중"

2) 문제 조건

입력 : 키 h와 몸무게 w가 공백을 기준으로 실수로 입력된다. ($1 \leq w, h \leq 200$)

출력 : 비만도 등급을 출력한다.

```
while(True):
    height = int(input("키를 입력하시오. >>> "))
    if(height >= 200):
        print("다시 입력해주세요.")
        continue
    else:
        break
while(True):
    weight = int(input("몸무게를 입력하시오. >>> "))
    if(weight <= 1):
        print("다시 입력해주세요.")
        continue
    else:
        break

if(height < 150):
    aveweight = height - 100
elif(height >= 150 and height<160):
    aveweight = ((height - 150) / 2) + 50
elif(height >= 160):
    aveweight = (height - 100) * 0.9

bmi = ((weight - aveweight) * 100) / aveweight

if(bmi <= 10):
    print("정상입니다.")
elif(bmi > 10 and bmi <= 20):
    print("과체중입니다.")
elif(bmi > 20):
    print("비만입니다.")
```

3-15 / 서윤철 / 202032029

1) 문제 설명

1~30 번중 번호를 추첨하여 상품을 주는 프로그램 입니다.

2) 문제 조건

- ① 1 ~ 30 까지 추첨 번호 생성.
- ② 1 등부터 4 등까지 상품을 주되 1等は 1명 2等は 3명 3等は 5명 4等は 6명 랜덤으로 지급
- ③ 중복 당첨은 배제.
- ④ 배열과 Random, if, elif, while 를 이용

```
from random import randint

fperson = 0
sperson = 0
tperson = 0
fourperson = 0
fcount = 0
scount = 0
tcount = 0
fourcount = 0

first = randint(1,30)
if(first == fperson):
    fcount += 1
if(fcount == 0):
    print("1 등 당첨자 없음.")
else:
    print("1 등 ", fcount, "명 ", " 상품 : 자동차")

for i in range(0,2):
    second = randint(1,30)
    if(second == first):
        second = randint(1,30)
    if(second == sperson):
        scount += 1
    if(scount == 0):
        print("2 등 당첨자 없음.")
    else:
        print("2 등 ", scount, "명 ", " 상품 : 바이크")

for i in range(0,4):
    third = randint(1,30)
    if(third == first or third == second):
        third = randint(1,30)
    if(third == tperson):
        tcount += 1
    if(tcount == 0):
        print("3 등 당첨자 없음.")
```

```
    else:
        print("3 등 ", tcount, "명 ", " 상품 : 킥보드")

for i in range(0,5):
    fourth = randint(1,30)
    if(fourth == first or fourth == second or fourth == third):
        fourth = randint(1,30)
    if(fourth == fourperson):
        fourcount += 1
    if(fourcount == 0):
        print("4 등 당첨자 없음.")
    else:
        print("4 등 ", fourcount, "명 ", " 상품 : 자전거")
```

3-16 / 김주현 / 201932013

1) 문제 설명

블랙잭의 설명

- ① 플레이어는 카드 두 장을 받는다. (이때, 두카드의 합이 21을 초과할 경우 패배한다.)
- ② 플레이어는 Hit(카드 추가)또는 Stand(승부)를 할 수 있다, Hit 를 하는 경우, 카드의 합이 21을 초과할 경우 패배한다.
- ③ 승부 시 플레이어의 카드 합이 딜러의 카드 합보다 높다면 승리한다. 만약 플레이어의 카드합이 딜러의 카드합보다 낮을 경우 패배한다.

2) 문제 조건

- ① 임의의 카드를 받아야한다.
- ② 받은 카드를 합산한 점수를 보여야한다.
- ③ while 문과 if 문을 활용 해야한다.
- ④ 승리, 비김, 패배 3 가지 경우가 있다.

```
from random import randint
player_first_card = randint(1,11)
player_second_card = randint(1,11)
player_hit = randint(1,11)

dealer_first_card = randint(1,11)
dealer_second_card = randint(1,11)

print("나의 카드는 : ", player_first_card, player_second_card)
if(player_first_card + player_second_card > 21):
    print("카드의 합이 21을 초과했습니다 .. 패배입니다. ")
elif(dealer_first_card + dealer_second_card > 21):
    print("딜러의 카드 합이 21을 초과하였습니다. 당신의 승리입니다 !")
elif(player_first_card + player_second_card == dealer_first_card + dealer_second_card):
    print("양 측의 카드 합이 모두 21을초과하였으므로 무승부입니다.")
else:
    answer = input("Hit / Go")
    if(answer == "Hit" or answer == "hit"):
        print("나의 카드는 : ", player_first_card, player_second_card, player_hit)
    elif(answer == "Go" or answer == "go"):
        print("Stand !")
        if(dealer_first_card + dealer_second_card > player_first_card + player_second_card):
            print("딜러의 카드는", dealer_first_card, dealer_second_card, "였습니다. 패배입니다 ..")
        elif(dealer_first_card + dealer_second_card < player_first_card + player_second_card):
            print("딜러의 카드는", dealer_first_card, dealer_second_card, "였습니다. 승리입니다 !")
```

3-17 / 용호중 / 201932033

1) 문제 설명

이 문제는 사용자인 user 와 컴퓨터인 com 이 단판 가위바위보를 하는 프로그램입니다

2) 문제 조건

- ① 컴퓨터가 가위 바위 보 중에 랜덤으로 하나를 내야 하므로 랜덤함수가 있어야 하고
- ② 이기거나 지거나 비기는 조건이 있어야 하므로 조건문도 필요합니다
- ③ 그리고 가위 바위 보를 리스트에 넣어 랜덤으로 꺼낼 때 편하게 만들었습니다

```
from random import randint
game = ["가위", "바위", "보"]

player_choice = input(" ( 가위 / 바위 / 보 ) ")

game_choice = randint(0,2)

if(game_choice == 0):
    if(player_choice == "가위"):
        print("무승부입니다.")
    elif(player_choice == "바위"):
        print("이겼습니다 !")
    elif(player_choice == "보"):
        print("패배입니다.")

if(game_choice == 1):
    if(player_choice == "가위"):
        print("패배입니다")
    elif(player_choice == "바위"):
        print("무승부입니다.")
    elif(player_choice == "보"):
        print("이겼습니다 !")

if(game_choice == 2):
    if(player_choice == "가위"):
        print("이겼습니다 !")
    elif(player_choice == "바위"):
        print("패배입니다 .")
    elif(player_choice == "보"):
        print("무승부입니다.")
```

3-18 / 이영민 / 201932044

1) 문제 설명:

로또 번호 6 개를 입력 받아 당첨을 확인하는 프로그램입니다.

2) 문제 조건:

- ① 난수를 생성하여 리스트에 저장해주세요.
- ② 내 번호 6 개를 입력 받아 리스트에 저장해주세요.
- ③ 맞은 개수를 6 개 1 등 5 개 2 등 순으로 출력해주세요.

```
from random import randint
count = 0
lotto = []
mylotto = []

for i in range(1,7):
    print("나의 ",i, "번째 예상 로또", end=" ")
    a = int(input("번호는 ? >>> "))
    mylotto.append(a)

for i in range(1,7):
    b = randint(0,45)
    lotto.append(b)

for i in range(0,6):
    if(mylotto[i] in lotto):
        print(i+1,"번째 숫자 ", lotto[i], "정답 !")
        count += 1
    elif(mylotto[i] not in lotto):
        print(i+1, "번째 숫자 ", lotto[i], "꽂 !")

if(count == 6):
    print("1 등 !")
elif(count == 5):
    print("2 등 !")
elif(count == 4):
    print("3 등 1")
else:
    print("다음 기회를 노려보세요.")
```


3-19 / 김동영 / 201932003

1) 문제 설명

파이썬으로 틱택토를 구현하시오.

2) 문제 조건

- ① def 를 이용해 함수를 사용한다
- ② 컴퓨터의 입력은 랜덤 모듈로 처리한다
- ③ 이차원 리스트를 활용한다

```
import random
board = [[0 for col in range (3)] for row in range (3)]
def printBoard():
    for i in board:
        for j in i:
            print(j,end='')
        print()

def isPossible(board):
    isPossible = False

if board == 0:
    isPossible=True
    return isPossible

def isWin(board,mark):
    isWin=False

if((board[0][0] == mark and board[0][1] == mark and board[0][2] == mark)or
   (board[1][0] == mark and board[1][1] == mark and board[1][2] == mark)or
   (board[2][0] == mark and board[2][1] == mark and board[2][2] == mark)or

   (board[0][0]==mark and board[1][0] == mark and board[2][0] == mark)or
   (board[0][1]==mark and board[1][1] == mark and board[2][1] == mark)or
   (board[0][2]==mark and board[1][2] == mark and board[2][2] == mark)or

   (board[0][0] == mark and board[1][1] == mark and board[2][2] == mark)or
   (board[0][2] == mark and board[1][1] == mark and board[2][0] == mark) ):
    isWin=True
    return isWin

def isEmpty(board):
    isEmpty= False

for i in board:
    for j in i:
        if(j==0):
            isEmpty=True
            return isEmpty
```

```

while(True):
    printBoard()
    x = int(input("표기할 곳을 고르세요 (0~2 세로축) :"))
    y = int(input("표기할 곳을 고르세요 (0~2 가로축) :"))
    if((x<0 or x>=3) or (y<0 or y>=3)) :
        print("잘못 입력되었습니다.")
        continue

    if isPossible(board[x][y]):
        board[x][y]='P'
    else:
        print("이미 놓여진 곳입니다.")
        continue

    if(isWin(board, 'P')):
        printBoard()
        print("플레이어 승리")
        break

    if (isEmpty(board)==False):
        printBoard()
        print("무승부!")
        break

    while(True):
        x=random.randint(0,2)
        y=random.randint(0,2)
        if(isPossible(board[x][y])):
            board[x][y]='X'
            break;

    if(isWin(board, 'X')):
        printBoard()
        print("컴퓨터 승리")
        break

```

3-20 / 김주혁 / 201932012

1) 문제 설명

키와 몸무게를 입력받아 키에 대한 정상체중을 구하고 bmi 수치를 계산해주는 프로그램입니다.

2) 문제 조건

- ① 남자, 여자에 따른 정상체중을 구하는 함수와 bmi 수치를 구하는 식의 함수가 있어야 한다
- ② 키는 m 단위로 입력받는다.
- ③ if else 문을 사용하여 출력
- ④ bmi 소수점은 2 자리까만 출력
- ⑤ 정상체중소수점은 1 자리까만 출력

```
height = float(input("키를 미터 단위로 입력하세요. >>>"))
weight = int(input("몸무게를 입력하세요. >>>"))

a = weight / height / height
c = height * height * 22
d = height * height * 21

if(a < 18.5):
    print("남성 표준 체형 : ",round(c,1), "여성 표준 체형 : ",round(d,1),"이며, 당신의 bmi 는",round(a,2)," , 저체중 입니다.")

elif(a > 18.5 and a <= 23):
    print("남성 표준 체형 : ",round(c,1), "여성 표준 체형 : ",round(d,1),"이며, 당신의 bmi 는",round(a,2)," , 정상 입니다.")

elif(a > 23 and a <= 25):
    print("남성 표준 체형 : ",round(c,1), "여성 표준 체형 : ",round(d,1),"이며, 당신의 bmi 는",round(a,2)," , 과체중 입니다.")

elif(a > 25 and a <= 30):
    print("남성 표준 체형 : ",round(c,1), "여성 표준 체형 : ",round(d,1),"이며, 당신의 bmi 는",round(a,2)," , 비만 입니다.")

else:
    print("남성 표준 체형 : ",round(c,1), "여성 표준 체형 : ",round(d,1),"이며, 당신의 bmi 는",round(a,2)," , 고도비만 입니다.")
```

3-21 / 우민우 / 201932034

1) 문제 설명

어떤 통화로 환전할 지 선택한 후 환전할 금액을 입력하여 환율을 계산하는 환율 계산기 프로그램입니다.

2) 문제 조건

- ① def 함수 이용
- ② if-else 문을 이용
- ③ while 문을 이용
- ④ 환율 기준은 인터넷 참고

```
answer = input("어떤 통화로 환전하시겠습니까 ? ( 달러 / 엔화 / 위안 )")

if(answer == "달러"):
    krw = int(input("환전할 금액을 입력하십시오. >>> "))
    usd = krw / 1426
    print(krw,"원을 환전하면 ", usd,"달러 입니다.")
if(answer == "엔화"):
    krw = int(input("환전할 금액을 입력하십시오. >>> "))
    jpy = krw / 9.8
    print(krw,"원을 환전하면 ", jpy,"엔 입니다.")
if(answer == "위안"):
    krw = int(input("환전할 금액을 입력하십시오. >>> "))
    cny = krw / 199.34
    print(krw,"원을 환전하면 ", cny,"위안 입니다.")
```

3-22 / 이유진 / 201932045

1) 문제 설명

숫자를 입력받아 소인수 분해를 하는 프로그램을 작성하세요.

2) 문제 조건

- ① 소인수한 값들을 전부 출력할것.
- ② 입력값이 소수라면 소수라고 출력 해야할것.

```
num = int(input("소인수 분해를 할 자연수를 입력하시오. >>> "))

result = []
i = 1

if(num < 1):
    print("1 이상의 값을 입력하시오.")
else:
    while(i <= num):
        if(num % i == 0):
            result.append(i)
            num = num / i
            print(i)
            i += 1

    if(len(result) == 2):
        print("입력하신 값은 소수입니다.")
```

3-23 / 김진용 / 201932014

1) 문제 설명

문제는 남성 20명 키를 배출하여 운동종목을 결정하는 프로그램을 작성하시오.

2) 문제 조건

- ① 키는 155~190 사이의 난수로 합니다.
- ② 185 이상 남자를 농구를 시키고 160 이하 남자를 달리기를 배정해줍니다.
- ③ 그 외에 남자를 줄다리기, 피구, 배구, 수영에 랜덤으로 값을 배정해줍니다.
- ④ 각 운동 종목에 몇명이 있는지 출력을 해줍니다.

```
from random import randint
```

```
work = []  
basketball = 0  
run = 0  
jul = 0  
p = 0  
base = 0  
swim = 0
```

```
for i in range(0,20):  
    a = randint(155,190)  
    work.append(a)
```

```
for i in range(0,20):  
    if(work[i] >= 185):  
        basketball += 1  
    elif(work[i] <= 160):  
        run += 1  
    else:  
        ran = randint(1,4)  
        if(ran == 1):  
            jul += 1  
        elif(ran == 2):  
            p += 1  
        elif(base == 3):  
            base += 1  
        elif(swim == 4):  
            swim += 1
```

```
print("농구 : ",basketball, "명")  
print("달리기 : ",run, "명")  
print("줄다리기 : ",jul, "명")  
print("피구 : ",p, "명")  
print("배구 : ",base, "명")  
print("수영 : ",swim, "명")
```

3-24 / 이세찬 / 201932041

1) 문제 설명

1~100 까지의 랜덤수 중 수를 몇번만에 맞추는지 보는 프로그램입니다

2) 문제 조건

- ① 난수 발생과 while 문을 이용하시오
- ② 수를 맞추면 종료되도록 구현하시오

```
from random import randint

AI = randint(1,100)
count = 1

while(True):
    answer = int(input("1 부터 100 까지 랜덤한 숫자가 입력되었습니다. 맞춰 보세요. >>> "))

    if(answer == AI):
        print("정답입니다 ! ", count, "번 만에 정답을 맞추셨습니다.")
        break
    else:
        if(answer < AI):
            print("오답입니다. 더 큰 수를 입력하셔야 합니다.")
            count += 1
        elif(answer > AI):
            print("오답입니다. 더 작은 수를 입력하셔야 합니다.")
            count += 1
```

3-26 / 정성민 / 201932053

1) 문제 설명

1 학기때 배운 스택(Stack)자료구조를 파이썬으로 간단하게 구현하고 삽입된 자료들을 오름차순 및 내림차순으로 정렬하세요.

2) 문제 조건

- ① 1.삽입 2. 반환 3. 공간늘리기 4. 오름차순 5. 내림차순 6. 끝내기 << 순으로 총 6 가지를 구현해 내세요.
- ② 스택 자료 내부의 아무 값이 들어가 있지 않은 곳은 '0'으로 채워 넣으세요.
- ③ 자료가 들어갈 리스트의 크기를 사용자에게 직접 입력받으세요.
- ④ 3 번 공간늘리기는 현재 리스트의 크기의 2 배씩 늘리도록 구현하세요.

```
num = int(input("스택의 크기를 입력하시오."))
stack = ["0"] * num
front = int(-1)
stack_length = len(stack)

while(True):

    menu = int(input(" ( 1.삽입 / 2.반환 / 3.공간 늘리기 / 4.오름차순 / 5.내림차순 / 6.종료 )"))
    if(menu == 1):
        #삽입
        if(front != stack_length - 1):
            num = int(input("넣을 자료를 입력하시오. >>> "))
            front += 1
            stack[front] = num
        else:
            print("공간이 부족합니다.")
            continue
        for i in range(0,stack_length):
            print(stack[i])
    elif(menu == 2):
        #반환
        if(front != -1):
            pop = stack[front]
            print(pop, " 반환 완료")
            stack[front] = '0'
            front -= 1
        for i in range(0,stack_length):
            print(stack[i])
    elif(menu == 3):
        #공간 늘리기
        stack = stack * 2
        for i in range(stack_length,stack_length * 2):
            stack[i] = '0'
            stack_length = len(stack)
        for i in range(0,stack_length):
            print(stack[i])
```



```

elif(menu == 4):
    #오름차순 정렬
    print("오름차순으로 정렬합니다.")
    for i in range(0,stack_length):
        if(stack[i] == "0"):
            break
        for j in range(0,stack_length):
            if(stack[j] == "0"):
                break
            if(stack[i] < stack[j]):
                temp = stack[i]
                stack[i] = stack[j]
                stack[j] = temp
    for i in range(0,stack_length):
        print(stack[i])
elif(menu == 5):
    #내림차순 정렬
    print("오름차순으로 정렬합니다.")
    for i in range(0,stack_length):
        if(stack[i] == "0"):
            break
        for j in range(0,stack_length):
            if(stack[j] == "0"):
                break
            if(stack[i] > stack[j]):
                temp = stack[i]
                stack[i] = stack[j]
                stack[j] = temp
    for i in range(0,stack_length):
        print(stack[i])
elif(menu == 6):
    break
#종료

```

1) 문제 설명

영어 점수(eng)와 수학 점수(math)를 입력받은 뒤 아래 공식을 참고하여 공분산과 상관계수를 구하는 프로그램을 완성하시오.

$$\text{Cov}(X, Y) = E((X - \mu_X)(Y - \mu_Y))$$

X, Y 확률변수

$\mu_X = E(X), \mu_Y = E(Y)$ X 와 Y 의 기댓값

$\text{Cov}(X, Y)$ X 와 Y 의 공분산

공분산 공식

$$\rho_{X,Y} = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}$$

$$r = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sqrt{\sum (x - \bar{x})^2 \times \sum (y - \bar{y})^2}}$$

X, Y 확률변수, σ_X, σ_Y X 와 Y 의 표준편차

\bar{x}, \bar{y} X 와 Y 의 표본평균

$\text{Cov}(X, Y)$ X 와 Y 의 공분산

$\rho_{X,Y}$ X 와 Y 의 모상관계수

r X 와 Y 의 표본상관계수

상관계수 공식

2) 문제 조건

① 영어 점수와 수학 점수의 개수(n)는 같다.

② for 문을 사용한다.

```

eng = list(map(int,input("영어 점수를 입력하시오. >>> ").split()))
math = list(map(int,input("수학 점수를 입력하시오. >>> ").split()))

n = len(eng)

sum_eng = sum(eng)
sum_math = sum(math)

sum_engmath = 0
for i in range(0,n):
    sum_engmath += eng[i] * math[i]

sum_eng2 = 0
for i in eng:
    sum_eng2 += i ** 2
sum_math2 = 0
for i in math:
    sum_math2 += i ** 2

Cov = round(sum_engmath / n - (sum_eng / n) * (sum_math / n),3)

A = round(sum_eng2 - (sum_eng) ** 2 / n,3)
B = round(sum_math2 - (sum_math) ** 2 / n,3)
C = round(sum_engmath - (sum_eng * sum_math) / n,3)
Rengmath = C / ((A ** 0.5) * (B ** 0.5))

print("공분산 : ",Cov)
print("상관계수 : ",Rengmath)

```

3-28 / 석상훈 / 201932028

1) 문제 설명

1 부터 50 까지의 숫자 중 랜덤으로 숫자 하나를 정답으로 정하고, 컴퓨터가 선택한 숫자와 사용자가 입력한 숫자를 비교하여 정답과 더 가까운 쪽이 승리하는 게임을 만들어보세요!

2) 문제 조건

- ① if 문을 활용하시오
- ② rand 함수 정수로 출력

```
from random import randint

result = randint(1,50)

player = int(input("숫자를 선택하십시오. "))
com = randint(1,50)

if(result - player > result - com):
    print("패배입니다.")
elif(result - player < result - com):
    print("승리입니다 !")
elif(result - player == result - com):
    print("무승부")
elif(player == result):
    print("승리입니다 !")
elif(com == result):
    print("패배입니다.")
```

3-29 / 이성훈 / 201832062

1) 문제 설명

두 사람이 주사위를 던져, 결과 보여주고 벌칙 수행 후 재도전 기회를 주고 포기 할 때 까지 대결하는 프로그램

2) 문제 조건

- ① 반복문 사용
- ② 동점시 재실행
- ③ random 사용하여 난수 생성

```
from random import randint

while(True):
    while(True):
        player1 = randint(1,6)
        player2 = randint(1,6)
        if(player1 == player2):
            print("Player 1 : ",player1, "Player 2 : ", player2)
            print("무승부이므로 재시작합니다.")
            continue
        if(player1 > player2):
            print("Player 1 : ",player1, "Player 2 : ", player2)
            print("Player 1의 승리, Player 2는 벌칙을 수행하십시오.")
            answer = input("재도전 ? ( Yes / No )")
            if(answer == "Yes"):
                continue
            elif(answer == "No"):
                break
        elif(player1 < player2):
            print("Player 1 : ",player1, "Player 2 : ", player2)
            print("Player 2의 승리, Player 1는 벌칙을 수행하십시오.")
            answer = input("재도전 ? ( Yes / No )")
            if(answer == "Yes"):
                continue
            elif(answer == "No"):
                break
```

3-30 / 박현식 / 201932027

1) 문제 설명

10000 원이 있는 player 가 농구관람료를 얻기 위해선 20000 원이 필요하다.

이때 참참참 게임을 해 이기면 3000 원, 지면 4000 원을 잃는다.

게임을 10 번 진행하여 농구관람료를 얻어라.(단, 돈이 없거나 20000 원 이상을 획득하면 게임은 종료한다.)

2) 문제 조건

- ① 보유금액이 20000 원 이상 또는 0 원 이하 일때 게임횟수에 상관없이 게임을 종료 시킬 것
- ② 게임이 10 번 진행되면 게임을 종료 시킬 것
- ③ random 모듈과 반복문 사용

```
from random import randint

player = 10000
count = 0

while(True):
    ccc = randint(1,2)
    if(ccc == 1):
        player += 3000
        print("Win !")
        print("현재 보유금액 = ",player)
    elif(ccc == 2):
        player -= 4000
        print("Lose")
        print("현재 보유금액 = ",player)

    if(player >= 20000):
        print("승리 !")
        break

    if(player <= 0):
        print("패배.")
        break

    if(count == 10):
        print("패배.")
        break

    count += 1
```

3-31 / 김민수 / 201932006

1) 문제 설명

플레이어와 컴퓨터는 랜덤한 수를 1~10 까지 출력하여 UP or DOWN 을 통해 두 수를 비교해 플레이어의 수가 더 높을 지, 낮을 지 예측하는 게임 프로그램

2) 문제 조건

- ① random 을 사용해 1~ 10 까지 정수인 난수를 생성
- ② while, if 반복문 사용
- ③ 1 또는 2 가 아닌 다른 수를 입력했을 시 다시 실행하게 구현

```
from random import randint

AI = randint(1,10)
count = 1

while(True):
    answer = int(input("1 부터 10 까지 랜덤한 숫자가 입력되었습니다. 맞춰 보세요. >>> "))

    if(answer == AI):
        print("정답입니다 ! ", count, "번 만에 정답을 맞추셨습니다.")
        break
    else:
        if(answer < AI):
            print("오답입니다. 더 큰 수를 입력하셔야 합니다.")
            count += 1
        elif(answer > AI):
            print("오답입니다. 더 작은 수를 입력하셔야 합니다.")
            count += 1
```

3-32 / 박상주 / 202121036

1) 문제 설명

문제는 나와 컴퓨터와의 가위바위보 6 판대결입니다

2) 문제 조건

- ① 반드시 반복문을 사용하여 진행합니다
- ② 또한 6 번의 가위바위보 대결중 비긴 결과도 그대로 출력합니다

```
from random import randint
i = 0

# 1 가위 2 바위 3 보
while(True):
    player = input("( 가위 / 바위 / 보 )")
    com = randint(1,3)

    if(player == "가위"):
        if(com == 1):
            print("무승부")
        elif(com == 2):
            print("패배")
        else:
            print("승!")

    if(player == "바위"):
        if(com == 1):
            print("승!")
        elif(com == 2):
            print("무승부")
        else:
            print("패배")

    if(player == "보"):
        if(com == 1):
            print("패배")
        elif(com == 2):
            print("승!")
        else:
            print("무승부")

    i += 1

    if(i == 6):
        break
```


3-33 / 최성현 / 202032054

1) 문제 설명

이름과 점수를 입력하면 학생의 이름과 점수에 따른 학점을 알려주는 프로그램 입니다.

2) 문제 조건

① while 문을 사용했습니다.

```
n = input("이름을 입력하십시오. >>> ")

while(True):
    score = int(input("점수를 입력하십시오. >>> "))
    if(score > 100):
        continue
    elif(score >= 90):
        print(n, " 학생의 성적은 A 입니다.")
    elif(score >= 80):
        print(n, " 학생의 성적은 B 입니다.")
    elif(score >= 70):
        print(n, " 학생의 성적은 C 입니다.")
    elif(score >= 60):
        print(n, " 학생의 성적은 D 입니다.")
    elif(score >= 50):
        print(n, " 학생의 성적은 E 입니다.")
    elif(score <= 40):
        print(n, " 학생의 성적은 F 입니다.")
```

4-01 / 전제민 / 201932052

1) 문제 설명

문제는 랜덤 영어(or 한국어) 단어 타자 연습 프로그램입니다.

2) 문제 조건

- ① 반드시 리스트를 사용
- ② 문제에 나오는 단어들이 랜덤하게 나올 것.
- ③ 문제는 1 번에서 7 번까지 나오게 구현
- ④ 타이핑한 단어가 맞으면 다음 문제로, 오타라면 다시 쓰게 구현

```
import random
word =
["Hello", "Welcome", "Python", "Netword", "Computer", "Software", "World", "Javascript", "Unity", "HTML"]
i = 1
while(True):
    if(i == 8):
        break
    word_random = random.choice(word)
    print(i, "번째 단어 : ", word_random)
    answer = input("단어를 입력하십시오. >>> ")
    if(answer == word_random):
        print("정답 !")
        i += 1
        continue
    else:
        while(True):
            answer = input("오답 ! 다시 입력하십시오.")
            if(answer == word_random):
                print("정답 !")
                i += 1
                break
```

4-02 / 이태호 / 201932048

1) 문제 설명

문제는 사용자의 과목별 성적을 입력받아 성적을 출력하는 프로그램입니다.

2) 문제 조건

- ① 반드시 튜플을 사용할 것
- ② for 문을 활용하여 key 값을 받을 것

```
oper = int(input('운영체제 점수 입력: '))
pyt = int(input('PYTHON 점수 입력: '))
pro = int(input('확률과 통계 점수 입력: '))
com = int(input('컴퓨터 구조 점수 입력: '))
java = int(input('JAVA Programming 점수 입력: '))
score = ({'운영체제':oper},{'PYTHON':pyt}, {'확률과 통계':pro},{'컴퓨터 구조':com},{'JAVA':java})
print(sco)
for item in score:
    for key in item.keys():
        if(item[key] >= 90):
            item[key] = 'A'
        elif(item[key] >= 80):
            item[key] = 'B'
        elif(item[key] >= 70):
            item[key] = 'C'
        elif(item[key] >= 60):
            item[key] = 'D'
        else:
            item[key] = 'F'
print(sco)
```

4-03 / 안명현 / 201932032

1) 문제 설명

csv 파일을 읽어들이고 뒤 json 형식으로 바꾸어 호출하는 get 방식 rest api 를 구축하고 싶다.
그러기 위해 필요한 csv to json 파서 모듈을 구현하시오.

2) 문제 조건

① file 오픈 함수를 통하여 csv 파일을 여시오

* open() 함수: 첫 번째 파라미터에 파일 주소, 두 번째 파라미터에 읽기 전용이라는 의미의 'r' 전달

* 읽어온 파일에서 readlines() 함수를 사용하여 파일을 한 줄씩 읽어온 리스트를 저장

* for line in lines: 구문으로 파일을 한 줄씩 읽어옴

② csv 파일은 콤마로 구분된 파일의 파일 형식이다. 문자열 파싱을 통하여 데이터를 나누시오

* CSV 파일의 첫 줄은 컬럼명이 적혀 있다.

③ json 파일 형식은 딕셔너리와 매우 유사하다

* 아래 형식은 json object 형식이다.

```
[  
{"name": "Ram", "email": "Ram@gmail.com"},  
{"name": "Bob", "email": "bob32@gmail.com"}  
]
```

* 리스트 안에 딕셔너리가 들어가 있는 형식과 유사하다.

④ 만약 데이터가 정수면 int 타입으로 형 변환하고 소수라면 float 타입으로 형 변환 하시오.

* try:

float(x) # 소수 타입이면 float 타입으로 형 변환

except ValueError:

pass # 문자열이라면 그대로 넘김

⑤ 읽어온 파일을 json 문자열로 바꾸어 출력하시오.

⑥ 테스트 csv 파일 첨부

```
CSV_FILE_PATH = "E:\\2022 년_과제\\my_python_questions\\컴퓨터네트워크문제만들기과제\\4 장\\data.csv" # 예제 파일 위치
```

```
csvFile = open(CSV_FILE_PATH, 'r')  
csvLines = csvFile.readlines()
```

```
jsonList = []
```

```
firstIndex = True
```

```
for line in csvLines:
```

```
    line = line.strip() # 줄바꿈 문자 제거
```

```
    if firstIndex == True: # csv 파일의 첫 번째 인덱스는 column 값이므로
```

```
        columnList = line.split(',') # 콤마를 기준으로 문자열을 자른 후 columnList 에 저장
```

```
        firstIndex = False # 수행 후 firstIndex 를 False 로 바꾸어 줌
```

```
    else:
```

```
        jsonDict = {}
```

```
valueList = line.split(',')
for i in range(len(columnList)):
    if valueList[i].isdigit(): # value 값이 숫자형이면 int 형으로 변환
        valueList[i] = int(valueList[i])
    else:
        try:
            valueList[i] = float(valueList[i])
        except ValueError:
            pass # 문자열이라면 아무것도 수행하지 않음

    jsonDict[columnList[i]] = valueList[i] # 딕셔너리에 값 추가

jsonList.append(jsonDict) # 리스트에 값 추가

print(jsonList) # jsonList 출력

csvFile.close() # 파일 닫기
```

4-04 / 이동근 / 201932039

1) 문제 설명

기온을 딕셔너리를 이용해 출력하고 가장 낮은 온도, 높은 온도, 30 도 이상인 날짜를 출력하는 문제입니다.

2) 문제 조건

- ① 기온을 딕셔너리를 이용해 출력할것
- ② 온도를 출력할때는 소숫점 까지 출력하게 할것
- ③ 날짜를 출력할때는 반드시 ,가 찍히게 할 것

```
temp = {'월':25.5, '화':28.3, '수':33.2, '목':32.1, '금':17.3, '토':35.3, '일':33.3}
lowtemp = temp.get("월")
hightemp = temp.get("월")
thtemp = temp.get("월")

for i in temp:
    print(i, end = ' : ')
    print(temp.get(i), "도")

for j in temp:
    if(temp.get(j) < lowtemp):
        lowtemp = temp.get(j)

    if(temp.get(j) > hightemp):
        hightemp = temp.get(j)

    if(temp.get(j) >= 30.0):
        thtemp = temp.get(j)
        print("30 도 이상 : ",thtemp)

print("최저기온 : ", lowtemp)
print("최고기온 : ", hightemp)
```

4-05 / 김채운 / 201932017

1) 문제 설명

비밀번호 오류시나 변경시 등장하는 보안 코드를 작성해보아라

2) 문제 조건

① random 함수를 사용할것 전역변수를 사용할 것

```
from random import randint

global code
code = randint(10000,99999)

print("출력되는 보안 코드를 그대로 입력하십시오.")

print("보안 코드 : ",code)

answer = int(input(">>> "))

if(answer == code):
    print("올바르게 입력하셨습니다.")
    quit()
else:
    print("잘못 입력하셨습니다. 다시 입력하십시오.")
```

4-06 / 장혁 / 201932051

1) 문제 설명

:휴대폰 전화부 친구이름을 추가, 삭제, 변경, 프로그램을 종료하는 프로그램을 출력하는 문제입니다.

2) 문제 조건

- ① 리스트를 이용할것.
- ② while,if,else 문을 이용할것.
- ③ 추가, 삭제, 변경, 종료 라는 단계가 있어야할 것.

```
phone = ["친구 1", "친구 2", "친구 3", "친구 4"]

while(True):
    select = int(input("( 1. 친구 리스트 출력 / 2.친구 추가 / 3.친구 삭제 / 4.이름 변경 / 5.종료 )"))
    if(select == 1):
        print(phone)
    if(select == 2):
        add = input("친구의 이름을 입력하시오. >>> ")
        phone.append(add)
    if(select == 3):
        delete = input("삭제할 친구의 이름을 입력하시오. >>> ")
        phone.remove(delete)
    if(select == 4):
        change = input("변경할 친구의 이름을 입력하십시오. >>> ")
        changed = input("어떻게 변경합니까 ? >>> ")
        phone.remove(change)
        phone.append(changed)
    if(select == 5):
        break
```


4-07 / 박예준 / 202032022

1) 문제 설명

영어 단어 퀴즈를 프로그램으로 작성하는 문제입니다.

2) 문제 조건

① 딕셔너리를 이용

② if ~elif else 문을 사용 해서 오답이면 다시풀기 다 풀면 처음부터 다시 하는 코드 작성

```
from random import randint
number = [1,2,3,4,5,6,7,8,9]

while(True):
    result = randint(5,9)
    a = int(input("a >>> "))
    b = int(input("b >>> "))
    c = int(input("c >>> "))

    if(a == b or a == c or b == c or a < b):
        print("다시 시도하십시오.")
        continue
    else:
        print("a * b + c = ",result)

    if(a * b + c == result):
        print("승리 !")
        break
    else:
        print("패배. 답은 다시 시도하십시오.")
```

4-08 / 박상수 / 201932025

1) 문제 설명

사용자가 소유하고 있는 돈과 고르고 싶은 주스를 입력하여 거스름돈을 출력하는 프로그램입니다.

2) 문제 조건

- ① 딕셔너리를 사용할 것
- ② 딕셔너리의 key,value 값을 리스트로 변형 시킬 것
- ③ 입력받은 돈,주스에 알맞는 거스름돈값을 출력 시킬것

```
slot = {"포도 주스":1700, "복숭아 주스":1000, "토마토 주스":1300, "알로에 주스":1200}
money = int(input("당신은 얼마의 돈을 가지고 있습니까 ? >>> "))
juice = input("무슨 주스를 구매하시겠습니까 ? ( 포도 주스 / 복숭아 주스 / 토마토 주스 / 알로에 주스 )>>> ")
remoney = 0

if(juice == "포도 주스"):
    remoney = money - slot.get("포도 주스")
    print("거스름돈은 ", remoney, "원 입니다.")
if(juice == "복숭아 주스"):
    remoney = money - slot.values("복숭아 주스")
    print("거스름돈은 ", remoney, "원 입니다.")
if(juice == "토마토 주스"):
    remoney = money - slot.values("토마토 주스")
    print("거스름돈은 ", remoney, "원 입니다.")
if(juice == "알로에 주스"):
    remoney = money - slot.values("알로에 주스")
    print("거스름돈은 ", remoney, "원 입니다.")
```

4-09 / 서윤철 / 202032029

1) 문제 설명

- 상품명과 가격을 입력 받아 사전에 각각 key와 value로 저장하는 프로그램을 구현.

< 출력 예시 >

첫 번째 상품명 : 냉장고

냉장고의 가격 : 2000000

두 번째 상품명 : 세탁기

세탁기의 가격 : 1000000

세 번째 상품명 : 에어컨

에어컨의 가격 : 1500000

{'냉장고': 2000000, '세탁기': 1000000, '에어컨': 1500000}

2) 문제 조건

① 딕셔너리 사용

② key, value 값을 리스트로 변형

③ 상품명과 가격을 입력해서 리스트로 출력

```
qlist = {}
pro = 0
praname = ""

for i in range(1,4):
    praname = input("제품 이름 >>> ")
    pro = int(input("제품 가격 >>> "))
    qlist[praname] = pro

qlist_key_list = list(qlist)
qlist_values_list = list(qlist.values())

for i in range(1,4):
    print(i, "번째 상품 : ", qlist_key_list[i-1], " / 가격 : ", qlist_values_list[i-1])
```

4-10 / 김현진 / 201932020

1) 문제 설명

집합이 같은 요소가 있는지 없는지 알아보아라.

2) 문제 조건

① `s1 = set([1, 2, 3, 4, 5])`

`s2 = set([4, 5, 6, 7, 8])`

`s3 = {1, 2, 3, 4, 5}`

`s4 = {6, 7, 8, 9, 10}` 를 사용하여라.

② `isdisjoint` 를 사용하여라.

③ `if else` 문을 사용하여라.

```
s1 = set([1,2,3,4,5])
s2 = set([4,5,6,7,8])
s3 = {1,2,3,4,5}
s4 = {6,7,8,9,10}

if s1.isdisjoint(s2):
    print("s1, s2 에는 같은 요소가 없습니다.")
else:
    print("s1, s2 에는 같은 요소가 하나 이상 있습니다.")
if s1.isdisjoint(s3):
    print("s1, s3 에는 같은 요소가 없습니다.")
else:
    print("s1, s3 에는 같은 요소가 하나 이상 있습니다.")
if s1.isdisjoint(s4):
    print("s1, s4 에는 같은 요소가 없습니다.")
else:
    print("s1, s4 에는 같은 요소가 하나 이상 있습니다.")
```

4-11 / 김진용 / 201932014

1) 문제 설명

문제는 성적을 입력하여 학점을 측정하는 프로그램을 만드시오

2) 문제 조건

1. 반드시 리스트를 활용
2. 조건문을 활용

```
grade = int(input("성적을 입력하십시오. >>> "))
gradelist = ["A", "B", "C", "D", "E"]

if(grade > 100 and grade >= 90):
    print(gradelist[0], " 학점 입니다.")

elif(grade > 90 and grade >= 80):
    print(gradelist[0], " 학점 입니다.")

elif(grade > 80 and grade >= 70):
    print(gradelist[0], " 학점 입니다.")

elif(grade > 70 and grade >= 60):
    print(gradelist[0], " 학점 입니다.")

else:
    print(gradelist[4], " 학점 입니다.")
```

4-12 / 김주현 / 201932013

1) 문제 설명

당첨자를 뽑는 프로그램을 만든다.

2) 문제 조건

- ① 신청한 사람마다 각각 정보(이름,전화번호,주소)가 저장되어야 한다.
- ② 당첨자는 랜덤으로 뽑아야한다.
- ③ 딕셔너리를 사용하고 저장된 값을 보여주어야한다.
- ④ 당첨된 사람의 이름을 보여줘야한다. (이때 이름의 값이 key, value 어디에 저장되어 있는지 주석으로 보이시오.)

```
from random import randint

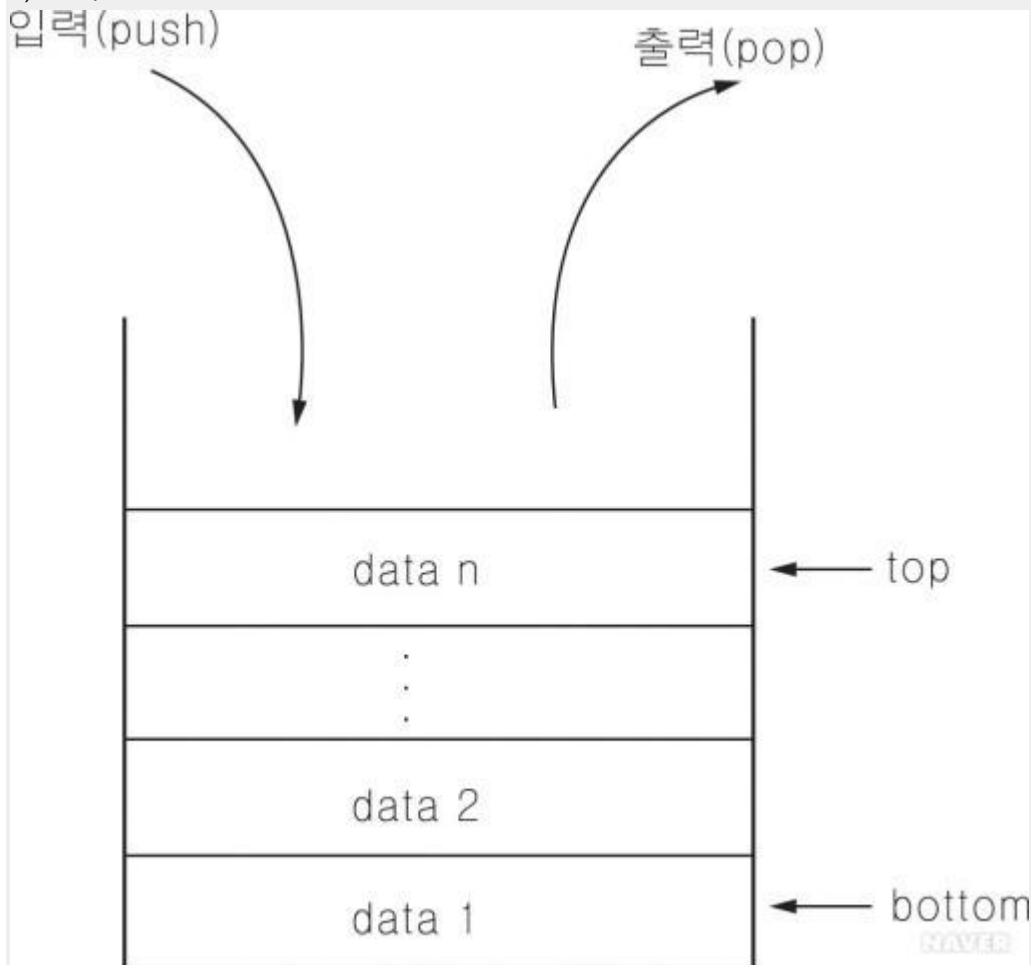
info = {}

i = 1

name = ""
number = ""
address = ""

while(True):
    win = randint(1,i)
    print(i, end = '')
    information = input("번째 신청자의 이름, 전화번호, 주소를 입력하십시오. >>> ").split()
    info[i] = [information]
    print(info)
    i += 1
    answer = input("더 입력합니까 ? ( Y / N )")
    if(answer == "Y"):
        continue
    elif(answer == "N"):
        for i in range(1,i):
            if(win == i):
                print(info[i], "당첨")
        break
```

1) 문제 설명



리스트를 이용하여 스택을 구현한 다음, 입력으로 주어지는 명령을 처리하는 프로그램을 작성하시오.

스택이란 모든 원소들의 삽입과 삭제가 리스트의 한쪽 끝에서만 수행되는 제한 조건을 가지는 선형 자료 구조로서, 삽입과 삭제가 일어나는 리스트의 끝을 top 이라 하고, 다른 한쪽 끝을 bottom 이라 한다.

스택의 top 에 새로운 원소를 삽입하는 것을 push 라 하고, 가장 최근에 삽입된 원소를 의미하는 스택의 top 으로부터 한 원소를 제거하는 것을 pop 이라 한다.

2) 문제 조건

① 다음 명령어들이 들어가야 한다.

push

데이터를 스택에 넣는 연산이다.

pop

스택에서 가장 위에 있는 데이터를 빼고, 그 수를 출력한다.

만약 스택에 들어있는 데이터가 없는 경우에는 데이터가 없다고 출력한다.

size

스택에 들어있는 데이터 수의 개수를 출력한다.

empty

스택이 비어있는지 확인한다.

top

스택의 가장 위에 있는 데이터를 출력한다.

만약 스택에 들어있는 데이터가 없는 경우에는 데이터가 없다고 출력한다.

② 다른 명령어들을 추가로 넣어도 된다.

③ 'stop'을 입력한 이후에는 현재 스택에 상황을 보여준다.

④ 사용자 정의 함수를 쓰지 않아도 된다.

```
stack = []

while(True):
    answer = input("수행할 연산을 입력하십시오. ( 조회 / push / pop / len / empty / top / stop ) >>> ")
    if(answer == "push"):
        push = input("넣을 값을 입력하십시오. >>> ")
        stack.append(push)

    elif(answer == "조회"):
        print(stack)

    elif(answer == "pop"):
        if(len(stack) == 0):
            print("스택이 공백 상태입니다.")
            continue
        else:
            print("스택의 가장 위에 있는 값은 ", stack.pop(), "입니다.")

    elif(answer == "len"):
        print("스택의 크기는 ", len(stack), "입니다.")

    elif(answer == "empty"):
        if(len(stack) == 0):
            print("스택이 공백 상태입니다.")
        else:
            print("스택에 값이 들어있는 상태입니다.")

    elif(answer == "top"):
        print("스택의 가장 위에 있는 값은 ", stack[len(stack)-1], "입니다.")

    elif(answer == "stop"):
        break
```


4-14 / 이유진 / 201932045

1) 문제 설명

첫번째 플레이어가 전화번호를 작성하면 두번째 플레이어는 최대한 적게 틀리고 숫자를 입력해 맞추는 프로그램입니다.

2) 문제 조건

- ① 몇번 틀렸는지 출력에 나오게 할것.
- ② 리스트를 사용해서 문제를 만들것.

```
player1 = ['_','_','_','_','_','_','_','_','_','_','_','_']
player2 = ['_','_','_','_','_','_','_','_','_','_','_','_']
fail = 0

for i in range(0,11):
    print(i+1,end = '')
    player1[i] = int(input("번째 자리의 전화번호를 입력하시오. >>> "))

while(True):
    for i in range(0,11):
        print(i+1,end = '')
        player2[i] = int(input("번째 자리의 전화번호를 맞춰보세요. >>> "))
    if(player1 == player2):
        print("정답을 맞추셨습니다! 시도 횟수 : ",fail)
        break
    else:
        print("틀리셨습니다.")
        fail += 1
```

4-16 / 이수현 / 201932043

1) 문제 구성

임의의 값을 보고 과연 어떤 값이 들어간지 추측해보자!

2) 문제 조건

- a,b,c 는 서로 다른 수 입니다
- a 는 b 보다 큰 수 입니다.

```
from random import randint
number = [1,2,3,4,5,6,7,8,9]

while(True):
    result = randint(5,9)
    a = int(input("a >>> "))
    b = int(input("b >>> "))
    c = int(input("c >>> "))

    if(a == b or a == c or b == c or a < b):
        print("다시 시도하십시오.")
        continue
    else:
        print("a * b + c = ",result)

    if(a * b + c == result):
        print("승리 !")
        break
    else:
        print("패배. 답은 다시 시도하십시오.")
```

4-16 / 김동영 / 201932003

1) 문제 구성

파이썬으로 끝말잇기 프로그램을 구현해 보시오.

2) 문제 조건

① 패배 조건은 다음과 같다.

-중복된 단어 사용

-바로 전 단어의 끝말과 다른 말로 시작하는 단어 사용

② 단어의 입력단위는 최소 2 개 이상이어야 한다.

③ 패배 조건은 함수로 구현해서 검사해야한다.

```
word_list = []
i = 0

def check(word):
    if(word_list[i][-1] != word[0]):
        return False

    elif(word in word_list):
        return False

    else:
        return True

word = input("Player 1의 차례입니다. 단어를 입력하십시오. 최소 단위 : 2글자 >>> ")
word_list.append(word)
print(word)

while(True):
    word = input("Player 2의 차례입니다. 단어를 입력하십시오. 최소 단위 : 2글자 >>> ")
    if(len(word) < 2):
        print("단어는 두 글자 이상이어야 합니다.")
        continue
    if(check(word) == True):
        word_list.append(word)
        print(word)
        i += 1
    else:
        print("패배입니다.")
        break

word = input("Player 1의 차례입니다. 단어를 입력하십시오. 최소 단위 : 2글자 >>> ")
if(len(word) < 2):
    print("단어는 두 글자 이상이어야 합니다.")
    continue
if(check(word) == True):
    word_list.append(word)
    print(word)
    i += 1
```

```
else:  
    print("패배입니다.")  
    break
```

4-17 / 김태균 / 202032012

1) 문제 구성

한쪽 끝이 막힌 주차장 프로그램 만들기 (스택구조)

2) 문제 조건

- ① 주차장에 들어갈 수 있는 최대 자동차 수는 5 대이다.
- ② 들어간 자동차의 이름과 현재 주차장의 상태를 출력한다.
- ③ 차량의 이름은 A, B, C, D, E 로 한다.

```
center = []

while(True):
    menu = input("작업을 선택하십시오. ( enter / exit / status / stop ) >>> ")
    if(menu == "enter"):
        push = input("어떤 자동차를 들어가게 하시겠습니까? ( A / B / C / D / E ) >>> ")
        if(push == "A"):
            center.append("A")
        elif(push == "B"):
            center.append("B")
        elif(push == "C"):
            center.append("C")
        elif(push == "D"):
            center.append("D")
        elif(push == "E"):
            center.append("E")

        if(len(center) == 5):
            print("자동차가 더 이상 들어갈 수 없습니다.")
            continue
    elif(menu == "exit"):
        center.pop(-1)
        print(center.pop(), " 자동차가 나갔습니다.")
        if(len(center) == 0):
            print("더 이상 나갈 자동차가 없습니다.")
    elif(menu == "status"):
        print(center)
    elif(menu == "stop"):
        break
```

4-19 / 고병준 / 201832001

1) 문제 설명

궁합이 좋은 음식을 추천해주고 그에 따른 음식 브랜드를 추천해주는 프로그램입니다.

2) 문제 조건

- ① 딕셔너리를 사용하리라.
- ② while 문을 사용하여 원하는 음식을 선택할 수 있도록 작성하리라.
- ③ input 함수를 사용하리라.
- ④ if else 함수를 사용하여 추천 브랜드를 작성하라.

```
food = {"떡볶이":"오뎅", "라면":"김치", "피자":"콜라", "치킨":"맥주", "돈까스":"우동"}
```

```
while(True):
    answer = input("음식을 입력하시오. >>> ")
    if(answer == "떡볶이"):
        print("추천 음식은 ",food.get("떡볶이"), " 입니다.")

    elif(answer == "라면"):
        print("추천 음식은 ",food.get("라면"), " 입니다.")

    elif(answer == "피자"):
        print("추천 음식은 ",food.get("피자"), " 입니다.")

    elif(answer == "치킨"):
        print("추천 음식은 ",food.get("치킨"), " 입니다.")

    elif(answer == "돈까스"):
        print("추천 음식은 ",food.get("돈까스"), " 입니다.")
```

4-20 / 이지민 / 201932047

1) 문제 설명

```
scores = ((2.8, 2.7), (3.9, 3.4))
```

- 한 학생의 학점이 다음과 같은 경우 평균학점 3.5 를 받기 위해 필요한 최소학점을 구하는 프로그램

2) 문제 조건

- ① for 문 사용하기
- ② 리스트, 튜플 사용하기
- ③ round 함수 사용하기

```
scores = ((2.8,2.7),(3.9,3.4))
score = 0
mul = 0

for i in range(0,2):
    j = 0
    mul = float(scores[i][j]) + float(scores[i][j+1])
    score = score + mul

a = 3.5 * 5
averagescore = a - score

print("받아야 할 학점은 ",averagescore, "입니다.")
```

4-21 / 이세찬 / 201932041

1) 문제 설명

값을 각각 key 와 value 에 저장한 딕셔너리를 리스트로 바꾸는 프로그램

2) 문제 조건

① 딕셔너리의 개수, 딕셔너리의 요소 수는 자유롭게 추가 할 수 있어야한다

```
dict_list = []
dictionary = {}
while(True):
    answer = input("( 생성 / 종료 )")
    if(answer == "생성"):
        key = input("키 입력 >>> ")
        value = input("값 입력 >>> ")
        dictionary[key] = value
        dict_list.append([key,value])

    if(answer == "종료"):
        print(dict_list)
```


4-22 / 최성현 / 202032054

1) 문제 설명

판매중인 과일을 볼 수 있고 추가적으로 빼거나 없앨수 있게 만든 프로그램입니다.

2) 문제 조건

① 리스트와 while, if else 문 사용

```
fruit = ["붉은 과일", "노란 과일", "검은 과일", "하얀 과일"]

while(True):
    menu = int(input("( 1.판매 중인 과일 / 2.새로 판매할 과일 / 3.판매 중지할 과일 )"))
    if(menu == 1):
        print(fruit)
    elif(menu == 2):
        new = input("과일의 이름을 입력하시오. >>> ")
        fruit.append(new)
    elif(menu == 3):
        rem = input("과일의 이름을 입력하시오. >>> ")
        fruit.remove(rem)
```

4-23 / 우민우 / 201932034

1) 문제 설명

예상 번호를 입력하여 로또 당첨 여부 확인

2) 문제 조건

- ① 리스트, 리스트 메소드 이용
- ② while 문 이용
- ③ if 문 이용
- ④ 난수 생성
- ⑤ not in 연산자 이용 중복여부 확인

```
from random import randint
lotto = []
mylotto = []

while(True):
    for i in range(0,6):
        lot = randint(1,45)
        if(lot not in lotto):
            lotto.append(lot)
        if(lot in lotto):
            continue

    for i in range(0,6):
        input1 = input("예상 로또 번호 6 개를 입력하시오. >>> ")
        mylotto.append(input1)

    for j in range(0,6):
        if(mylotto[j] in lotto):
            print(j,"정답")
        else:
            print(j,"꽁")
```

4-24 / 용호중 / 201932033

1) 문제 설명

자신의 통장에서 돈을 입금 출금 잔액부족이면 얼마가 부족한지를 알아내는 프로그램

2) 문제 조건

- ① 잔액은 random 함수 이용하여 쓰기 단, 잔액은 10000 원부터
- ② while 문을 이용하여 0 원이 되면 종료, 입출금을 그만하고 싶을 때 까지 반복하기
- ③ if 문을 사용하여 입출금 번호매기기

```
from random import randint

bank = randint(10000,100000)

print("현재 잔액은 ",bank, "원 입니다.")

while(True):
    menu = input("( 입금 / 출금 )")
    if(menu == "입금"):
        add = int(input("얼마를 입금하시겠습니까 ? >>> "))
        bank = bank + add
        print("입금완료, 현재 잔액은 ",bank,"원 입니다.")
    if(menu == "출금"):
        minus = int(input("얼마를 출금하시겠습니까 ? >>> "))
        if(bank >= minus):
            bank = bank - minus
            print("출금완료. 현재 잔액은 ",bank,"원 입니다.")
        else:
            print("출금불가, 현재 잔액은 ",bank,"원 이며 부족한 금액은 ",minus-bank, "원 입니다.")
            continue
```

1) 문제 설명

생태학에서 나무의 분포도를 측정하는 것은 중요하다. 그러므로 당신은 미국 전역의 나무들이 주어졌을 때, 각 종이 전체에서 몇 %를 차지하는지 구하는 프로그램을 만들어야 한다.

2) 문제 조건

① 프로그램은 여러 줄로 이루어져 있으며, 한 줄에 하나의 나무 종 이름이 주어진다. 어떤 종 이름도 30 글자를 넘지 않으며, 입력에는 최대 10,000 개의 종이 주어지고 최대 1,000,000 그루의 나무가 주어진다.

```
import sys

total = 0
dic = dict()
while(True):
    word = sys.stdin.readline().rstrip()
    if word == '':
        break
    total += 1
    if word in dic:
        dic[word] += 1
    else:
        dic[word] = 1
sdic = dict(sorted(dic.items()))
for i in sdic:
    a = sdic[i]
    per = ( a / total * 100)

    print(i, "%.", per * 4)
```

4-26 / 석상훈 / 201932038

1) 문제 설명

100 점 만점의 점수에 대해 각 등급의 개수를 카운트하는 프로그램이다.

2) 문제 조건

- ① while 문 이용
- ② 리스트 이용
- ③ 등급은 A,B,C,D,F

```
from random import randint

scorelist = []

for i in range(0,20):
    score = randint(0,100)
    scorelist.append(score)

a = 0
b = 0
c = 0
d = 0
f = 0

for j in range(0,20):
    if scorelist[j] >= 90 and scorelist[j] <= 100:
        a += 1

    elif scorelist[j] >= 80 and scorelist[j] <= 89:
        b += 1

    elif scorelist[j] >= 70 and scorelist[j] <= 69:
        c += 1

    elif scorelist[j] >= 60 and scorelist[j] <= 59:
        d += 1

    elif scorelist[j] >= 0 and scorelist[j] <= 59:
        f += 1

print("A : ",a,"명")
print("B : ",b,"명")
print("C : ",c,"명")
print("D : ",d,"명")
print("F : ",f,"명")
```

4-27 / 김민규 / 201932005

1) 문제 설명

점심 메뉴를 랜덤으로 고르기

2) 문제 조건

- ① 랜덤으로 뽑을 것
- ② 딕셔너리를 사용할 것
- ③ 메뉴와 전화번호가 같이 출력되기 할 것

```
from random import randint

menu = ["비빔밥", "순대국", "제육볶음", "라면", "먹지 않음"]
number = {"비빔밥" : "111-234", "순대국" : "222-345", "제육볶음" : "333-456", "라면" : "444-567",
"먹지 않음" : "없음"}

choice = randint(0, len(menu)-1)

choiced = menu[choice]
print(choiced, ":", number.get(choiced))
```

4-28 / 정성민 / 201932053

1) 문제 설명

한달 용돈 사용한 내역을 입력하면 다음달까지 남은 일 수와 하루평균 얼마를 썼는지를 계산해주는 프로그램

2) 문제 조건

- ① 딕셔너리를 통해 1~12 달까지 구현해서 사용할 것
- ② 사용한 내역이 차례대로 리스트안에 들어갈 수 있도록 구현할 것

```
cal =
{'1월':31,'2월':28,'3월':31,'4월':30,'5월':31,'6월':30,'7월':31,'8월':31,'9월':30,'10월':31,'11월':30,'12월':31}

month = input("현재 몇월입니까 ? ex)'3월' >>> ")
calvalue = int(cal.get(month))

money = int(input("얼마의 용돈을 받으니까 ? >>> "))
date = 1
count = 1

while(True):
    print(date,"일 입니다.")
    use = int(input("얼마의 용돈을 사용했습니까?"))
    if(use > money):
        print("다시 입력하십시오.")
        continue
    else:
        money = money - use
        print("현재 남은 용돈은 ",money,"원 입니다.")
        print("이번 달의 남은 일 수는 ",calvalue-date)
        if(calvalue == date or money <= 0):
            break
        date += 1

money = money / date
print("평균적으로 하루에 ",money / date, "만큼 용돈을 소비하셨습니다.")
```

4-29 / 박주혁 / 202032025

1) 문제 설명

문제는 무작위 한자리 수를 맞추는 프로그램입니다.

2) 문제 조건

- ① 리스트를 사용 할것
- ② 틀린 횟수를 출력에 나오게 할것

```
from random import randint

number = randint(1,9)

failcount = 0

while(True):
    answer = int(input("랜덤한 한 자리 숫자가 결정되었습니다. 정답을 맞춰 보세요. >>> "))
    if(answer == number):
        print("정답입니다 !")
        break
    else:
        failcount += 1
        print("오답입니다. 틀린 횟수 : ",failcount)
        continue
```


4-30 / 이영민 / 201932044

1) 문제 설명

아이스크림 자판기입니다.

2) 문제 조건

- ① 딕셔너리를 이용하여 아이스크림과 가격을 정할것
- ② 아이스크림마다 개수를 입력받아 총금액을 출력할것

```
ice = {'메로나':300, '비비빅':400, ' 죠스바':600, '수박바':500, '빵빠레':1000}

mcount = 0
bcount = 0
jcount = 0
wcount = 0
pcount = 0

while(True):
    choice = input("구매할 아이스크림을 선택하세요. ( 메로나 / 비비빅 / 죠스바 / 수박바 / 빵빠레 ) >>> ")

    if(choice == "메로나"):
        count = int(input("구매할 개수는 ? >>> "))
        mcount = ice['메로나'] * count

    elif(choice == "비비빅"):
        count = int(input("구매할 개수는 ? >>> "))
        bcount = ice['비비빅'] * count

    elif(choice == "죠스바"):
        count = int(input("구매할 개수는 ? >>> "))
        jcount = ice['죠스바'] * count

    elif(choice == "수박바"):
        count = int(input("구매할 개수는 ? >>> "))
        wcount = ice['수박바'] * count

    elif(choice == "빵빠레"):
        count = int(input("구매할 개수는 ? >>> "))
        pcount = ice['빵빠레'] * count

    break

print("총액은 ",mcount+bcount+jcount+wcount+pcount,"입니다.")
```

4-31 / 김주혁 / 201932012

1) 문제 설명

카페 키오스크 프로그래밍으로 주문을 받고 돈을 입력받아 계산하고 잔돈을 거슬러주는 프로그램입니다.

2) 문제 조건

- ① 리스트를 이용하여라
- ② while 문을 이용하여라
- ③ if 문을 이용하여라

```
cost = [4000,4500,5000,6000]
money = 0
count = 0

while(True):
    menu = input("메뉴를 선택하십시오. ( 버블티 / 아메리카노 / 라떼 / 에이드 / 주문완료 ) >>> ")
    if(menu == "버블티"):
        count = int(input("몇 잔 주문합니까 ? >>> "))
        money = money + cost[0] * count
        continue

    if(menu == "아메리카노"):
        count = int(input("몇 잔 주문합니까 ? >>> "))
        money = money + cost[1] * count
        continue

    if(menu == "라떼"):
        count = int(input("몇 잔 주문합니까 ? >>> "))
        money = money + cost[2] * count
        continue

    if(menu == "에이드"):
        count = int(input("몇 잔 주문합니까 ? >>> "))
        money = money + cost[3] * count
        continue

    if(menu == "주문완료"):
        print("총액은 ",money, "원 입니다.")
        break
```

4-32 / 김민수 / 201932006

1) 문제 설명

점심 혹은 저녁식사로 간단한 메뉴를 리스트 값에서 한 가지를 랜덤으로 출력해 사용자에게 추천하는 프로그램입니다.

2) 문제 조건

- ① 리스트를 사용한다.
- ② while 문과 if 문을 사용한다.

```
from random import randint
count = 0
lunch = ['제육볶음', '김치찌개', '만둣국', '김치볶음밥', '돈까스']
dinner = ['어묵탕', '순대국', '떡볶이', '자장면', '순두부찌개']

while(True):
    lunch1 = randint(0, len(lunch)-1)
    dinner1 = randint(0, len(dinner)-1)
    time = input("시간대 ? ( 점심 / 저녁 ) >>> ")
    if(time == "점심"):
        print("추천 메뉴는 ", lunch[lunch1], "입니다.")
    elif(time == "저녁"):
        print("추천 메뉴는 ", dinner[dinner1], "입니다.")
```

4-33 / 박현식 / 201932027

1) 문제 설명

자료구조를 이용해 사람들의 이름과 키를 가르쳐주고 직접 그 자릿값을 입력해 농구 포지션 추천을 해주는 프로그램입니다.

2) 문제 조건

- ① 자료구조는 딕셔너리를 이용할 것
- ② 조건문을 이용할 것

```
spec = {'Kim':165, 'Lee':175, 'Park':180, 'Han':185, 'Jeon':190, 'Ha':200}

name = input("정보를 얻고 싶은 사람의 이름을 입력하십시오. >>> ")
height = int(spec.get(name))

if(height >= 170 and height <= 180):
    print(name, "은 ", "guard(가드) 포지션에 어울립니다.")

if(height > 180 and height <= 190):
    print(name, "은 ", "forward(포워드) 포지션에 어울립니다.")

if(height > 190 and height <= 230):
    print(name, "은 ", "center(센터) 포지션에 어울립니다.")

else:
    print(name, "님은 다른 운동을 하는 것을 추천 드립니다.")
```

5-01 / 전제민 / 201932052

1) 문제 설명

문제는 함수를 이용한 계산기 프로그램입니다.

2) 문제 조건

- ① 반드시 함수를 사용할 것
- ② 기본적인 더하기, 빼기, 곱하기, 나누기 등의 기능을 필수적으로 넣을 것
- ③ 종료하지 않는 한 무한루프가 되게 프로그래밍 할 것

```
sum = lambda x,y : x+y
minus = lambda x,y : x-y
divide = lambda x,y : x/y
remainder = lambda x,y : x%y

def seek(cal):
    if(cal == "+" or cal == "-" or cal == "%" or cal == "/"):
        return True
    else:
        return False

answer = int(input("정수 계산기 프로그램입니다. 첫번째 수를 입력하십시오. "))
i = 1
while(True):
    print(i, "번째 연산입니다.", answer, end='')
    cal = input("에 하고 싶은 연산을 입력하십시오. ( + - / % ) >>> ")
    if(seek(cal) == False):
        print("잘못된 접근입니다.")
        break
    a = int(input("연산할 숫자를 입력하십시오. "))
    print("연산 결과: ",answer,cal,a,"=",end='')
    if(cal == "+"):
        answer = int(sum(answer,a))
    elif(cal == "-"):
        if(answer < a):
            answer = int(minus(a,answer))
            answer = -answer
        else:
            answer = int(minus(answer,a))
    elif(cal == "%"):
        answer = int(remainder(answer,a))
    elif(cal == "/"):
        if(answer <= 0):
            print("잘못된 접근입니다.")
        else:
            try:
                answer = int(divide(answer,a))
            except:
                print("0으로 나눌 수 없습니다.")
                break
    print(answer)
    ctn = input("연산을 계속 진행합니까? (Yes / No) >>> ")
```

```
if(ctn == "Yes"):
    i += 1
    continue
elif(ctn == "No"):
    break
else:
    break
```

5-02 / 이태호 / 201932048

1) 문제 설명

사용자에게 3 과목의 점수를 입력받아 총점과 평균, 등급을 계산하는 프로그램입니다.

2) 문제 조건

- ① 함수는 3 개 사용 할 것
- ② 입력 과목을 3 개이상을 할 것

```
def score():
    score = int(input("점수를 입력하십시오. >>> "))
    return score

def point(a):
    if(a >= 90):
        return "A"
    elif(a >= 80):
        return "B"
    elif(a >= 70):
        return "C"
    elif(a >= 60):
        return "D"
    else:
        return "F"

def total():
    print("총 점수 : ",sum, "평균 : ",average, "등급 : ",point(average))

operation = score()
python = score()
java = score()

sum = operation + python + java
average = int(sum / 3)

total()
```

5-03 / 이동근 / 201932039

1) 문제 설명

로또 번호를 랜덤으로 생성하는 프로그램입니다.

2) 문제 조건

- ① 반드시 함수를 사용할 것
- ② 로또 번호 6 개를 생성한다.
- ③ 로또 번호는 1~45 까지의 랜덤한 번호다
- ④ 6 개의 숫자 모두 달라야 한다
- ⑤ getRandomNumber() 함수를 사용해서 구현한다

```
from random import randint
lotto = []

def getRandomNumber():
    number = randint(1,45)
    return number

for i in range(0,6):
    rannum = getRandomNumber()
    if(rannum not in lotto):
        lotto.append(rannum)

print(lotto)
```


5-05 / 장혁 / 201932051

1) 문제 설명

bmi 를 구할수 있는 프로그램을 만들어 보세요.

2) 문제 조건

- ① def 함수를 이용할것
- ② 고도비만, 비만, 과체중, 정상체중 이렇게 4 가지의 경우의 수를 만들것
- ③ while 문을 사용할것
- ④ 프로그램을 또 이용할수 있도록 할것

```
def bmical(weight,height):  
    bmi = height - weight  
    if(bmi > 30):  
        result = "고도비만"  
    elif(bmi >= 25):  
        result = "비만"  
    elif(bmi >= 15):  
        result = "과체중"  
    else:  
        result = "정상체중"  
    return result  
  
while(True):  
    weight = int(input("체중 입력 >>> "))  
    height = int(input("키 입력 >>> "))  
    result = bmical(weight,height)  
  
    print(result)
```

5-06 / 김현진 / 201932020

1) 문제 설명

함수를 이용해서 계산기를 만들어 보아라.

2) 문제 조건

- ① 함수를 사용하여라.
- ② 연산기호 + - // *를 사용하여라
- ③ if elif 사용

```
sum = lambda x,y : x+y
sub = lambda x,y : x-y
mul = lambda x,y : x*y
div = lambda x,y : x/y

x = int(input("x 입력 >>> "))
y = int(input("y 입력 >>> "))
cal = input("연산 기호 입력 ( + / - * / / ) >>> ")
if(cal == "+"):
    a = sum(x,y)
elif(cal == "-"):
    a = sub(x,y)
elif(cal == "*"):
    a = mul(x,y)
elif(cal == "/" or cal == "//"):
    a = div(x,y)

print(a)
```

5-07 / 박상수 / 201932025

1) 문제 설명

사용자에게 달을 입력받아 계절을 출력하는 프로그램입니다.

2) 문제 조건

- ① 함수를 사용할 것
- ② 달을 13 이상 입력시 오류로 출력시킬 것

```
def month1(num):  
    if(num == 3 or num == 4 or num == 5):  
        return 1  
    elif(num == 6 or num == 7 or num == 8):  
        return 2  
    elif(num == 9 or num == 10 or num == 11):  
        return 3  
    elif(num == 12 or num == 1 or num == 2):  
        return 4  
  
while(True):  
    month = int(input("몇 월의 계절을 알고 싶으십니까 ? >>> "))  
    if(month <= 12):  
        if(month1(month) == 1):  
            print("봄")  
        elif(month1(month) == 2):  
            print("여름")  
        elif(month1(month) == 3):  
            print("가을")  
        elif(month1(month) == 4):  
            print("겨울")
```

5-08 / 박예준 / 202032022

1) 문제 설명

주사위 2개 6회 던져서 합이 41 이상 나오면 "win", 그렇지 않으면 "Lose"

도중에 10, 20, 30, 40 이면 "Lose" Double (같은 수)이 나오면 기회 추가 프로그램 입니다.

2) 문제 조건

- ① random 모듈의 choice 함수를 이용
- ② 도중에 10, 20, 30, 40 나오면 종류
- ③ 주사위 1 과 주사위 2 가 같은수 나오면 한번더

```
from random import randint
sum = 0

while(True):
    for i in range(0,6):
        dice1 = randint(1,6)
        dice2 = randint(1,6)
        print("1:",dice1,"2:",dice2)
        sum = sum + dice1 + dice2
        if(sum == 10 or sum == 20 or sum == 30 or sum == 40 or dice1 == dice2):
            print("보너스 기회 +1 !")
            continue

    if(sum >= 41):
        print("Win !")
        break

    elif(sum <= Lose):
        print("Lose !")
        break
```

5-09 / 김주현 / 201932013

1) 문제 설명

도시의 이름을 영어로 입력받고 출력하는 프로그램을 만드시오.

2) 문제 조건

① 함수를 사용하시오.

② 메뉴를 만들어서 추가, 확인, 종료를 선택할 수 있는 기능을 넣으시오.

```
def menu():
    print("( 1.도시 추가 / 2.확인 / 3.종료 )")

city_list = []

while(True):
    menu()
    a = int(input())

    if(a == 1):
        city = input("도시 이름을 영어로 입력하십시오. >>> ")
        city_list.append(city)
    elif(a == 2):
        print(city_list)
    elif(a == 3):
        break
```

5-10 / 이유진 / 201932045

1) 문제 설명

1~50 중 몇개의 폭탄을 랜덤으로 설정하고 컴퓨터와 플레이어가 번갈아가며 선택하게 하여 지뢰를 피하는게임입니다.

(윈도우에 있는 지뢰찾기와 비슷한 게임입니다.)

2) 문제 조건

- ① 폭탄의 개수를 직접 입력하게 할 것.
- ② 함수를 만들어 사용할 것.

```
from random import randint
field = []
bomb = 0
randombomb = []
i = 0

bombcount = int(input("폭탄의 개수를 설정하십시오. 1~50 >>> "))

for k in range(0,50):
    field.append("")

for i in range(0,50):
    rand = randint(0,1)
    if(rand == 1):
        field[i] = rand
        bomb += 1
        if(bomb < bombcount):
            for j in range(0,50):
                if(field[j] == ""):
                    rand = randint(0,1)
                    if(rand == 1):
                        field[j] = rand
                        bomb += 1
                        if(bomb == bombcount):
                            break
            if(bomb == bombcount):
                break

while(True):
    print(i,end = '')
    player1 = input("Player 1 의 첫번째 시도 ( 0 / X ) >>> ")
    if(player1 == "0"):
        if(field[i] == ""):
            print("Player 1 패배.")
            break
        elif(player1 == "X"):
            if(field[i] != ""):
                print("Player 1 패배.")
                break
```

```
i += 1

print("Computer 의 ", i , "번째 차례.")
com = randint(0,1)
if(com == 0):
    if(field[i] == ""):
        print("Computer 패배.")
        break
elif(com == 1):
    if(field[i] != ""):
        print("Comnuper 패배.")
        break
```

5-11 / 이지민 / 201932047

1) 문제 설명

초기 투자액과 투자 기간 그리고 투자 기간 중 날짜별 가치 변동율을 입력 받아서 순이익과 이익 여부를 구하는 프로그램

2) 문제 조건

- ① 투자액은 100 이상 10000 이하의 정수입니다.
- ② 투자 기간은 1 이상 10 이하의 정수입니다.
- ③ 일별 변동폭은 -100 이상 100 이하의 정수로 주어집니다.

```
money = int(input("초기 투자금액 :"))
date = int(input("투자 기간 :"))
result = []
for i in range(date):
    print(i+1,end = "")
    changed = int(input("일차 수익 변동률 : "))
    result.append(changed)

def stock(a,d,p):
    inv = a
    for i in range(d):
        m = inv * int(p[i]) / 100
        inv += m
        print(i+1, "일차 : ",inv)
    return inv

print("결과 / ",end = "")
t=stock(money, date, result)
print("총금액 : ",t)

a = t - money
if (a > 0):
    print("올랐습니다!")
elif (a == 0):
    print("보통입니다.")
elif (a < 0):
    print("떨어졌습니다..")
```


5-12 / 이수현 / 201932043

1) 문제 설명

간단한 계산기 제작

2) 문제 조건

① 계산기에서 제시된 구하기는 전부 함수로 제작해야 할 것

```
sum = lambda x,y : x+y
sub = lambda x,y : x-y
mul = lambda x,y : x*y
div = lambda x,y : x/y

x = int(input("x 입력 >>> "))
y = int(input("y 입력 >>> "))
cal = input("연산 기호 입력 ( + / - * / / ) >>> ")
if(cal == "+"):
    a = sum(x,y)
elif(cal == "-"):
    a = sub(x,y)
elif(cal == "*"):
    a = mul(x,y)
elif(cal == "/"):
    a = div(x,y)

print(a)
```

5-13 / 김태균 / 202032012

1) 문제 설명

사용자가 값들을 입력하고 그 값들에 대한 합, 최댓값, 최솟값, 평균, 분산을 구하는 계산기 제작

2) 문제 조건

① 합, 최댓값, 최솟값, 평균, 분산은 전부 함수로 제작한다.

```
def Sum(num):
    Sum = sum(num)
    return Sum

def Max(num):
    Max = max(num)
    return Max

def Min(num):
    Min = min(num)
    return Min

def Average(num):
    ave = sum(num) / len(num)
    return ave

def Var(num):
    var = 0
    for i in num:
        var = var + (i - sum(num) / len(num)) ** 2
    return var

num = list(map(int, input("값을 입력하십시오. >>> ").split()))
while(True):
    menu = input("( 합 / 최댓값 / 최솟값 / 평균 / 분산 / 종료 ) >>> ")
    if(menu == "합"):
        Sum = Sum(num)
        print(Sum)
    elif(menu == "최댓값"):
        Max = Max(num)
        print(Max)
    elif(menu == "최솟값"):
        Min = Min(num)
        print(Min)
    elif(menu == "평균"):
        Average = Average(num)
        print(Average)
    elif(menu == "분산"):
        Var = Var(num)
        print(Var)
    elif(menu == "종료"):
        break
def Average(num):
    ave = sum(num) / len(num)
    return ave
```

```
def Var(num):
    var = 0
    for i in num:
        var = var + (i - sum(num) / len(num)) ** 2
    return var

num = list(map(int, input("값을 입력하십시오. >>> ").split()))
while(True):
    menu = input("( 합 / 최댓값 / 최솟값 / 평균 / 분산 / 종료 ) >>> ")
    if(menu == "합"):
        Sum = Sum(num)
        print(Sum)
    elif(menu == "최댓값"):
        Max = Max(num)
        print(Max)
    elif(menu == "최솟값"):
        Min = Min(num)
        print(Min)
    elif(menu == "평균"):
        Average = Average(num)
        print(Average)
    elif(menu == "분산"):
        Var = Var(num)
        print(Var)
    elif(menu == "종료"):
        break
```

5-14 / 김채운 / 201932017

1) 문제 설명

게임 플레이어 인터페이스를 만든다.

필요한것은 플레이어의 이름은 입력받고 스테이터스는 random 함수를 이용하여 생성하라

2) 문제 조건

- ① 함수를 매개 변수로 사용할 것
- ② 스테이터스 호출 제작

```
from random import randint

status1 = {'HP':0, 'Luk':0, 'Str':0, 'San':0, 'Int':0, 'Dex':0}

def status():
    status1['HP'] = randint(1,10)
    status1['Luk'] = randint(1,10)
    status1['Str'] = randint(1,10)
    status1['San'] = randint(1,10)
    status1['Int'] = randint(1,10)
    status1['Dex'] = randint(1,10)

name = input("캐릭터의 이름을 입력하십시오. >>> ")
status()
print(name, "의 스테이터스 : ", status1)
```

5-15 / 안명현 / 201932032

1) 문제 설명

파이썬 내장 함수인 sorted() 함수를 직접 구현하시오.

2) 문제 조건

- ① 첫번째 파라미터 : 리스트, 튜플, 문자열
- ② 두번째 파라미터 reverse : True 내림차순, False 오름차순, default = False
- ③ 리턴 값으로는 항상 List 로 반환
- ④ 파이썬 sorted 함수는 병합 정렬을 기반으로 만들어짐 (시간복잡도 $N\log N$)

```
# 병합 정렬 기반 sorted 함수, 정렬한 것을 반환
def Sorted(arr, reverse = False):
    if len(arr) < 2:
        return arr

    mid = len(arr) // 2
    low = Sorted(arr[:mid], reverse=reverse) # 재귀적으로 호출하며 리스트를 부분 리스트로 쪼갬
    high = Sorted(arr[mid:], reverse=reverse) # 분할정복법 : divide and conquer

    mergedArr = []
    l = 0
    h = 0
    while l < len(low) and h < len(high): # 2 개의 리스트를 비교하며 더 작거나 큰 값을 mergedArr 에
        # 추가
        if reverse:
            if low[l] > high[h]: # 내림차순일 경우
                mergedArr.append(low[l])
                l += 1
            else:
                mergedArr.append(high[h])
                h += 1
        else:
            if low[l] < high[h]: # 오름차순일 경우
                mergedArr.append(low[l])
                l += 1
            else:
                mergedArr.append(high[h])
                h += 1

    mergedArr += low[l:]
    mergedArr += high[h:]

    return mergedArr

# 테스트 코드
if __name__ == "__main__":
```

```
print("List Test")
l1 = [6, 2, 4, 1, 5]
l1 = Sorted(l1)
print(l1)

l2 = [6, 2, 4, 1, 5]
l2 = Sorted(l2, reverse=True)
print(l2)

print("Tuple Test")
t1 = (4, 2, 3, 1, 6)
t1 = Sorted(t1)
print(t1)

t2 = (4, 2, 3, 1, 6)
t2 = Sorted(t2, reverse=True)
print(t2)

print("String Test")
s1 = "68321"
s1 = Sorted(s1)
print(s1)

s2 = "68321"
s2 = Sorted(s2, reverse=True)
print(s2)
```

5-16 / 고병준 / 201832001

1. 문제 설명

영화와 뮤지컬중 선택하고 나이에 따른 표값을 출력하는 문제입니다.

2. 문제 조건

- ① 영화, 뮤지컬의 표값을 출력하는 옵션을 만들어라
- ② 나이에 따른 표값을 나눠라
- ③ 함수와 while 문을 사용하여라

```
def mfee(age):
    if(age >= 18):
        fee = 8000
    else:
        fee = 4000
    return fee

def mufee(age):
    if(age >= 18):
        fee = 12000
    else:
        fee = 6000
    return fee

age = int(input("당신의 나이는 ? >>> "))
choice = input("( 영화 / 뮤지컬 ) >>> ")

if(choice == "영화"):
    mchoice = input("( A / B / C ) >>> ")
    if(mchoice == "A"):
        print("가격 : ",mfee(age))
    elif(mchoice == "B"):
        print("가격 : ",mfee(age)+1000)
    elif(mchoice == "C"):
        print("가격 : ",mfee(age)+2000)

elif(choice == "뮤지컬"):
    muchoice = input("( A / B / C ) >>> ")
    if(muchoice == "A"):
        print("가격 : ",mufee(age))
    elif(muchoice == "B"):
        print("가격 : ",mufee(age)+1000)
    elif(muchoice == "C"):
        print("가격 : ",mufee(age)+2000)
```

5-17 / 최지훈 / 202032055

1) 문제 설명

파이썬을 이용하여 근의 공식을 구현해보아라.

2) 문제 조건

①a,b,c 의 값을 입력받아 해를 출력하라.(단 a는 0 이 아니다.)

만약 해의 값이 없다면 "허근"을 출력하라.

②format 문을 이용할 경우 {}안에 변수의 값을 넣을 수 있다.

ex) print("{}는 바보야 ".format(your_name))

③출력시 your_name 는 바보야

```
def cal(a,b,c):
    i = (b ** 2) - (4 * a * c)
    if(i > 0):
        x = (-b + i ** 0.5) / (2 * a)
        y = (-b - i ** 0.5) / (2 * a)
        print(a, "x^2 + ",b, "x +", c, "= +",x, "-", y)
    elif(i == 0):
        x1 = -b / 2 * a
        print("중근 = ",x1)
    else:
        print("허근")

a = int(input("a >>> "))
b = int(input("b >>> "))
c = int(input("c >>> "))

cal(a,b,c)
```


5-18 / 김동영 / 201932003

1) 문제 설명

그림을 그릴수 있는 프로그램을 만드시오. 단 출력은 상하반전된 그림을 포함해야 한다.

2) 문제 조건

① 좌표입력과 그림 출력은 함수로 작성한다

```
SIZE = 10

board = [['□' for i in range(SIZE)] for i in range(SIZE)]
def showBoard():
    for i in range(SIZE):
        for j in range(SIZE):
            print(board[i][j],end = '')
        print()

def showReversedBoard():
    for i in range(SIZE):
        for j in range(SIZE):
            print(board[SIZE-(i+1)][SIZE-(j+1)],end = "")
        print()

def boardChange(x,y):
    if(board[y][x] == '■'):
        board[y][x] = '□'
    elif(board[y][x] == '□'):
        board[y][x] = '■'

while(True):
    showBoard()
    x = int(input("도트를 찍을 X 좌표 입력. >>> "))
    y = int(input("도트를 찍을 Y 좌표 입력. >>> "))
    if(int(input("입력을 계속하려면 0 입력 (종료는 0 제외 아무 숫자.) >>> "))):
        break
    boardChange(x,y)

    showBoard()
    print("상하좌우 반전")
    showReversedBoard()
```

5-19 / 이영민 / 201932044

1) 문제 설명

시험성적을 입력받아 등급을 알려주는 프로그램

2) 문제 조건

- ① 이름을 입력받아 출력할것
- ② 각 과목별로 함수를 만들어서 출력할것.

```
def korean(score):
    if(score >= 90):
        print("국어 성적은 1 등급입니다.")
    elif(score >= 80):
        print("국어 성적은 2 등급입니다.")
    elif(score >= 70):
        print("국어 성적은 3 등급입니다.")
    elif(score >= 60):
        print("국어 성적은 4 등급입니다.")
    elif(score >= 50):
        print("국어 성적은 5 등급입니다.")
    elif(score >= 40):
        print("국어 성적은 6 등급입니다.")
    elif(score >= 30):
        print("국어 성적은 7 등급입니다.")
    elif(score >= 20):
        print("국어 성적은 8 등급입니다.")
    elif(score >= 0):
        print("국어 성적은 9 등급입니다.")

def english(score):
    if(score >= 90):
        print("영어 성적은 1 등급입니다.")
    elif(score >= 80):
        print("영어 성적은 2 등급입니다.")
    elif(score >= 70):
        print("영어 성적은 3 등급입니다.")
    elif(score >= 60):
        print("영어 성적은 4 등급입니다.")
    elif(score >= 50):
        print("영어 성적은 5 등급입니다.")
    elif(score >= 40):
        print("영어 성적은 6 등급입니다.")
    elif(score >= 30):
        print("영어 성적은 7 등급입니다.")
    elif(score >= 20):
        print("영어 성적은 8 등급입니다.")
    elif(score >= 0):
        print("영어 성적은 9 등급입니다.")

def korean(score):
    if(score >= 90):
```

```

        print("국어 성적은 1 등급입니다.")
    elif(score >= 80):
        print("국어 성적은 2 등급입니다.")
    elif(score >= 70):
        print("국어 성적은 3 등급입니다.")
    elif(score >= 60):
        print("국어 성적은 4 등급입니다.")
    elif(score >= 50):
        print("국어 성적은 5 등급입니다.")
    elif(score >= 40):
        print("국어 성적은 6 등급입니다.")
    elif(score >= 30):
        print("국어 성적은 7 등급입니다.")
    elif(score >= 20):
        print("국어 성적은 8 등급입니다.")
    elif(score >= 0):
        print("국어 성적은 9 등급입니다.")

def math(score):
    if(score >= 90):
        print("수학 성적은 1 등급입니다.")
    elif(score >= 80):
        print("수학 성적은 2 등급입니다.")
    elif(score >= 70):
        print("수학 성적은 3 등급입니다.")
    elif(score >= 60):
        print("수학 성적은 4 등급입니다.")
    elif(score >= 50):
        print("수학 성적은 5 등급입니다.")
    elif(score >= 40):
        print("수학 성적은 6 등급입니다.")
    elif(score >= 30):
        print("수학 성적은 7 등급입니다.")
    elif(score >= 20):
        print("수학 성적은 8 등급입니다.")
    elif(score >= 0):
        print("수학 성적은 9 등급입니다.")

name = input("이름을 입력하십시오. >>> ")
korean1 = int(input("국어 성적 >>> "))
english1 = int(input("영어 성적 >>> "))
math1 = int(input("수학 성적 >>> "))

print(name, "님의 등급")
korean(korean1)
english(english1)
math(math1)

```

5-20 / 김민규 / 201932005

1) 문제 설명

기초 대사량을 구할 수 있는 프로그램을 만들어 보시오.

2) 문제 조건

① if문을 사용 할 것, 남 여 구분을 하여 서로 다른 값을 출력하게 할 것.

```
age = int(input("나이 >>> "))
sex = input("( Male / Female ) >>> ")
weight = int(input("몸무게 >>> "))
height = int(input("키 >>> "))

if(sex == "Male"):
    bmr = 66.5 + (13.75 * weight) + (5 * height) - (6.755 * age)
elif(sex == "Female"):
    bmr = 65.51 + (9.6 * weight) + (1.8 * height) - (4.7 * age)

print(bmr)
```

5-21 / 이세찬 / 201932041

1) 문제 설명

로또 번호를 출력하는 프로그램

2) 문제 조건

- ① getRandomNumber 함수로 랜덤 수를 리턴

```
from random import randint
lotto = []

def getRandomNumber():
    number = randint(1,45)
    return number

for i in range(0,6):
    rannum = getRandomNumber()
    if(rannum not in lotto):
        lotto.append(rannum)

print(lotto)
```

5-22 / 용호중 / 201932033

1) 문제 설명

사람들의 하루 권장 카페인 섭취량과 현재까지 먹은 양에서 오늘 남은 하루 권장 카페인양을 알기 위해

성인 청소년 임산부 중 하나를 입력받고 현재까지 섭취한 카페인 양을 빼서 남은 권장 카페인 양을 알기위한 프로그램

2) 문제 조건

- ① 값을 반환하는 함수로 caffeine 이라는 함수를 쓸것
- ② 조건문 이용

```
def caffeine(a):
    if(a == "성인"):
        c = 400
    elif(a == "청소년"):
        c = 200
    elif(a == "임산부"):
        c = 300
    else:
        print("잘못 입력하셨습니다.")
    return c

a = input("( 성인 / 청소년 / 임산부 ) >>> ")
caf = int(input("금일 섭취한 카페인 총량을 입력하십시오. (커피 한잔에 95mg) >>> "))

print(a,"의 카페인 권장량은 ",caffeine(a) , "이며, 금일 추가 섭취해도 괜찮은 카페인 양은 ", caffeine(a)
- caf )
```

5-23 / 우민우 / 201932034

1) 문제 설명

회원가입과 로그인이 가능한 프로그램

2) 문제 조건

- ① 컨테이너
- ② 반복문
- ③ 조건문
- ④ 리스트

```
join = input("( 회원가입 / 로그인 ) >>> ")
id=[]
pw=[]
a=[]
i=0
while(join == "회원가입"):
    identity = input("아이디를 입력하십시오. >>> ")
    for member in id:
        while(member == identity):
            print(identity, "중복된 아이디")
            identity = input("다시 입력하십시오.")
    password = input("비밀번호를 입력하십시오. >>> ")
    id.append(identity)
    pw.append(password)
    print("회원가입이 완료 되었습니다.")
    join = input("( 회원가입 / 로그인 )")

entryid = input("아이디를 입력하십시오. >>>")
for mem in id:
    if(mem == entryid):
        a.append(i)
    i = i + 1

while(len(a) == 0):
    print("아이디가 존재하지 않습니다.")
    entryid = input("아이디를 다시 입력하십시오. >>> ")
    i = 0
    for mem in id:
        if(mem == entryid):
            a.append(i)
            i = i + 1
entrypw = input("비밀번호를 입력하십시오. >>>")

while(entrypw != pw[a[0]]):
    print("비밀번호가 일치하지 않습니다.")
    entrypw = input("비밀번호를 다시 입력하십시오. >>>")
print("로그인 되었습니다.")
```

5-24 / 나성진 / 202032017

1) 문제 설명

N 장의 카드가 있다.

각각의 카드는 차례로 1 부터 N 까지의 번호가 붙어 있으며, 1 번 카드가 제일 위에, N 번 카드가 제일 아래인 상태로 순서대로 카드가 놓여 있다.

이제 다음과 같은 동작을 카드가 모두 버려질 때 까지 반복한다.

1. 제일 위에 있는 카드를 바닥에 버린다.
2. 다음, 제일 위에 있는 카드를 제일 아래에 있는 카드 밑으로 옮긴다.

2) 문제 조건

- ① 정수 N 을 입력한다.
- ② 버리는 카드들과 카드들의 순서를 순서대로 출력한다.

```
card = []
count = 0

N = int(input("카드는 몇장 ? >>> "))

for i in range(1,N+1):
    card.append(i)

print(card.pop(0), "(" ,1, ")")

for j in range(1,N+1):
    if(len(card) == 0):
        break
    card.append(card[0])
    card.pop(0)
    print(card.pop(0), "(" ,j+1, ")")
```


5-25 / 김진용 / 201932014

1) 문제 설명

음식점에 입장해 메뉴를 고르는 간단한 프로그램

2) 문제 조건

① def 함수를 사용

```
def food(menu):
    if(menu == "라면" or menu == "햄버거"):
        price = 5000
    elif(menu == "만두" or menu == "감자튀김"):
        price = 10000
    else:
        price = 2000
    return price

menu = input("먹고 싶은 메뉴를 선택하십시오. ( 라면 / 햄버거 / 치킨 / 피자 ) >>> ")
print("가격 : ", food(menu))
```

5-26 / 정성민 / 201932053

1) 문제 설명

랜덤으로 받은 수를 내림차순으로 직접 정렬해보세요

2) 문제 조건

- ① 함수를 사용해서 수를 랜덤으로 받아오세요
- ② 랜덤으로 수를 받아서 저장하는 리스트는 전역변수로 선언해주세요

```
from random import randint

global randsu
randsu = []
global randsu2
randsu2 = []

def rand():
    for i in range(0,10):
        randsu.append(randint(1,10))

def sort():
    i = 0
    while(i<11):
        if(i in randsu):
            for j in range(0,10):
                if(i == randsu[j]):
                    randsu2.append(i)
            i += 1

rand()
sort()

print(randsu2)
```

5-27 / 박현식 / 201932027

1) 문제 설명

나이, 키, 체중을 입력받아 체질량지수(bmi)를 측정하는 프로그램을 작성하라.

2) 문제 조건

- ① 함수를 사용할 것.
- ② 지역 변수를 사용할 것.
- ③ 제어문을 사용할 것.

```
def bmical(height, weight):
    if(height < 150):
        aveweight = height - 100
    elif(height >= 150 and height<160):
        aveweight = ((height - 150) / 2) + 50
    elif(height >= 160):
        aveweight = (height - 100) * 0.9
    return aveweight

while(True):
    age = int(input("나이를 입력하십시오. >>> "))
    height = int(input("키를 입력하십시오. >>> "))
    if(height >= 200):
        print("다시 입력해주세요.")
        continue
    else:
        break

while(True):
    weight = int(input("몸무게를 입력하십시오. >>> "))
    if(weight <= 1):
        print("다시 입력해주세요.")
        continue
    else:
        break

aveweight = bmical(height, weight)

bmi = ((weight - aveweight) * 100) / aveweight

if(bmi <= 10):
    print("정상입니다.")
elif(bmi > 10 and bmi <= 20):
    print("과체중입니다.")
elif(bmi > 20):
    print("비만입니다.")
```

5-28 / 김민수 / 201932006

1) 문제 설명

간단한 구구단을 출력하는 프로그램

2) 문제 조건

① def 함수 사용

② for, if, while 문 사용

```
def multiple(a):  
    for i in range(a,a+1):  
        for j in range(1,10):  
            print(i, " * ",end = " ")  
            print(j, " = ", i*j)  
  
a = int(input("몇 단 출력 ? >>> "))  
multiple(a)
```

5-29 / 김주혁 / 201932012

1) 문제 설명

채널과 볼륨을 입력받아 줄이고높이는 리모컨 프로그램을 만들어보세요

2) 문제 조건

- ① 채널과 볼륨을 낮추고 높이는 함수를만들어라
- ② while 문과 if 문을 사용하여라
- ③ 올리고 낮추기를 할때 값을 계속 저장하게 하여라

```
def channelcontrol(count):
    channel = 0
    channel = count
    print("현재 채널 : ", channel)

def volumecontrol(count):
    volume = 0
    volume += count
    if(volume <= 0):
        print("음소거 되었습니다.")
        volume = 0
    elif(volume >= 100):
        print("최대 볼륨입니다.")
        volume = 100
    else:
        print("현재 볼륨 : ", volume)

while(True):
    menu = int(input("( 1.볼륨 조정 / 2.채널 조정 / 3.전원 끄기 ) >>> "))
    if(menu == 1):
        up = int(input("볼륨을 얼마나 올리시겠습니까 ? ( + / - ) >>> "))
        volumecontrol(up)
    elif(menu == 2):
        cha = int(input("몇 번 채널로 이동합니까 ? >>> "))
        channelcontrol(cha)
    elif(menu == 3):
        quit()
```

5-30 / 석상훈 / 201932028

1) 문제 설명

세 수 중 가장 큰 수, 가장 작은 수 찾기

2) 문제 조건

- ① 함수를 이용할 것
- ② 수를 입력 받아야함

```
def maxseek(a,b,c):
    if(max(a,b,c) == a):
        print("가장 큰 수는 ",a, "입니다. ")
    elif(max(a,b,c) == b):
        print("가장 큰 수는 ",b, "입니다. ")
    elif(max(a,b,c) == c):
        print("가장 큰 수는 ",c, "입니다. ")

def minseek(a,b,c):
    if(min(a,b,c) == a):
        print("가장 작은 수는 ",a, "입니다. ")
    elif(min(a,b,c) == b):
        print("가장 작은 수는 ",b, "입니다. ")
    elif(min(a,b,c) == c):
        print("가장 작은 수는 ",c, "입니다. ")

a = int(input("첫 번째 수 입력 >>> "))
b = int(input("두 번째 수 입력 >>> "))
c = int(input("세 번째 수 입력 >>> "))

maxseek(a,b,c)
minseek(a,b,c)
```

결과 검토

여러 문제들이 공통되거나 비슷한 형식으로 구성되어 있었으며, 대부분 저의 수준으로 구현 가능한 프로그램이었으나, 평균적인 실력에 비해 뛰어난 몇몇의 학생들 또한 분명히 존재하였고, 그들의 존재는 같은 학생으로써 제가 목표로 삼을 수 있는 기준점 중 가장 가까운 곳에 있는 존재라 생각합니다.

특히, 안명현 학생의 문제가 전부 인상 깊었는데, 그가 제출한 3 장 과제 (자동화 PVP 시스템 구현) 를 풀어보며 정말 많은 시간이 걸렸고, 그 만큼 많은 시행착오를 겪었지만,

해당 문제를 직접 풀어냈을 때의 성취감, 쾌감은 이루 말할 수 없을 정도였습니다. 이러한 성취감이 공부를 더 열심히 할 수 있게 하는 원동력이 되는 것 같습니다.

정리하여,

본 과제를 수행하며, 반복적인 프로그램 제작을 통해 여러 능력을 향상시킬 수 있었으나 아직 프로그램 구현 시 시간 단축, 불필요한 코드 간소화 등의 능력을 많이 키워야겠다는 생각이 들었습니다. 다음 학기에는 사전 준비를 철저히 하여 좋은 성적을 노려보고자 합니다.

감사합니다.

-마침-