

2023년도 2학기 “소프트웨어적사고” 기말고사 live coding

2023. 12. 7. (목)

이름 :

학번 :

■ 기말고사 라이브코딩 관련 공지사항

- 총 160점
- 시험시간 : 2023. 12. 7(목) 18:00 ~ 20:30
- 장소 : 미래관 445
- Off-line / Open book

- 제출 관련

o 작성된 프로그램과 캡처한 실행화면을 ecampus로 기말고사 제출 link로 upload 하기 바랍니다.

o 파일명을 반드시 다음과 같이 사용하십시오. (반드시 Idle 환경에서 통합실행 가능해야 합니다.)

프로그램 화일명 : ISSN_driver.py
ISSN.py

생성된 화일명 : input_issn_list.txt
certify_issn_number.pkl

실행화면 캡처화일 : Final_학번_이름_캡처.png

o 종료시간 이후에는 형평성을 위해 프로그램 upload 절대 불가능 합니다. (ecampus에 upload 마감시간 setting). 시험 이후 이메일 제출 절대 불가능합니다. (제출해도 평가하지 않습니다).

- 주의사항

o 시험시간 동안 자리를 이석하지 마시기 바랍니다.

o Cheating 절대금지 : 아래의 경우 본 과목 F로 처리

- 1) 남의 프로그램을 표절하거나 남에게 위탁하여 코딩을 하는 행위
- 2) 제시된 logic을 이용하여 문제를 해결하지 않고 출력문 등을 사용하여 프로그램 입출력형식만을 보여주는 등 fake 행위 등 편법을 사용하는 것.

- 3) 일반적인 모든 경우를 다루는 데이터에 적용되는 프로그램을 작성해야 합니다. 주어진 예제 데이터 만으로 한정지어 성공/실패를 확인하도록 코딩을 하는 행위는 금지됩니다. (예 : 주어진 7개의 검증 data만을 7개의 if문으로 모두 하나씩 검증하는 행위 등)
- 4) Hard coding 절대금지
- 5) AI 등의 도움을 얻어 coding하는 경우
- 6) 이외 컴퓨터공학도 및 미래의 전문직 엔지니어로서 양심에 반하는 행위 등

- 채점 기준

- o idle 환경사용
- o **Error 발생하여 실행이 안되면 무조건 0점.**
- o Error 없이 실행이 되는 경우에만 문제에서 제시한 논리 및 조건 (공통조건 포함)에 따라 채점 진행

- 다음의 선서문을 제출하는 프로그램의 첫 번째 라인에 주석문으로 올리기를 바랍니다.

'''

선서 : 나는 양심에 따라 절대 부정행위를 하지 않겠습니다. 부정행위 적발 시 모든 민사, 형사상의 법적책임을 지겠습니다. 이름 홍길동 (20233456), 2023년 12월 7일 (목)

'''

[2023년도 2학기 “소프트웨어적 사고” 기말고사 live coding 문제지]

■ 공통조건

- Class만을 사용하시오. (이 조건을 만족하지 않을 경우 전체 0점 처리)
- 생성자(constructor)를 사용하시오.
- 소멸자(destructor)를 사용하시오.
- User-defined module 사용 (모든 파일을 동일 folder(directory)에 저장하시오.)
- 모든 프로그램에 system-defined exception handler 사용. 다만 3)단계 에서는 user defined exception handler를 사용하시오.
- Main class를 driver로 정의.
- file open / close를 담당하는 method를 분리하시오.
- Comment를 충분히 넣으시오.
- Echo checking을 반드시 하시오.
- 다음의 프로그램 입출력형식을 따르시오.

```
>> input_issn_list.txt에서 ISSN 번호를 차례로 읽어옵니다.
>> 파일에서 읽어온 내용 : ['00279358\n', '20493630\n', '0000300X\n', '0057935X\n', '08912350\n', '0891235K\n', 'TTTTT']
>> ISSN code 검증을 시작합니다.

>> 1번째 ISSN code : 00279358
>> ISSN 코드 00279358은(는) 검증되었습니다.

>> 2번째 ISSN code : 20493630
>> ISSN 코드 20493630은(는) 검증되었습니다.

>> 3번째 ISSN code : 0000300X
>> ISSN 코드 0000300X은(는) 검증되었습니다.

>> 4번째 ISSN code : 0057935X
>> ISSN 코드 0057935X은(는) 검증되지 않았습니다.

>> 5번째 ISSN code : 08912350
>> ISSN 코드 08912350은(는) 검증되지 않았습니다.

>> 6번째 ISSN code : 0891235K
>> Exception) 잘못된 코드. 마지막 문자가 X가 아닙니다.

>> 7번째 ISSN code : TTTTT
>> Exception) 잘못된 코드. 숫자나 X이외의 문자를 포함하고 있습니다.

>> 생성된 Dictionary입니다 : {'00279358': 'RIGHT', '20493630': 'RIGHT', '0000300X': 'RIGHT', '0057935X': 'WRONG', '08912350': 'WRONG', '0891235K': 'WRONG', 'TTTTT': 'WRONG'}

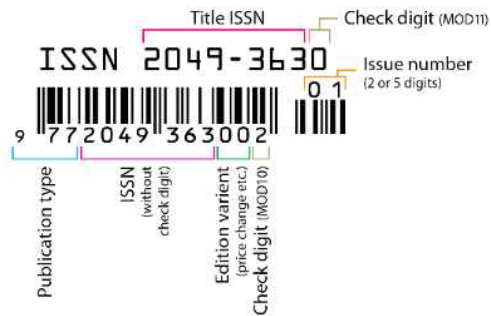
>> Pickling을 시작합니다.
>> Dump : certify_issn_number.pkl 파일을 생성했습니다.

>> Load : certify_issn_number.pkl 파일을 읽어옵니다.
>> Unpickled 결과입니다 : {'00279358': 'RIGHT', '20493630': 'RIGHT', '0000300X': 'RIGHT', '0057935X': 'WRONG', '08912350': 'WRONG', '0891235K': 'WRONG', 'TTTTT': 'WRONG'}
```

■ 문제 : ISSN 번호 검증

[ISSN 번호검증 알고리즘]

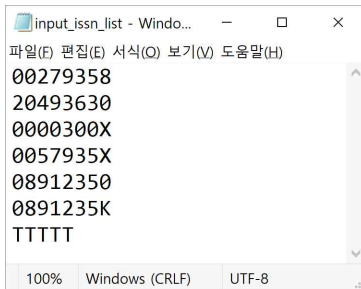
- ISSN은 국제 표준 연속 간행물 번호(International Standard Serial Number, ISSN)는 인쇄물이나 정기적 전자 간행물을 식별하는 데 쓰이는 8자리 고유 번호임



- 위의 그림에서 보는 바와 같이 ISSN code는 8자리로 구성 (Title ISSN + Check digit)
예) 2049-3630
- 마지막 자리(왼쪽부터 8번째 자리, 앞의 예에서 마지막 붉은색 글씨 0)는 check digit
- Check digit는 다음과 같이 산출한다.
Step 1) 왼쪽부터 7자리 숫자에 8부터 2까지의 숫자를 순서대로 곱하여 모두 더한 값으로 가중치를 구한다.
예) $(2*8) + (0*7) + (4*6) + (9*5) + (3*4) + (6*3) + (3*2) = 121$
Step 2) 가중치를 11로 나누어 나머지를 구한다.
예) $121 / 11 \rightarrow$ 나머지 0
Step 3) 11에서 Step 2에서 구한 나머지를 뺀다. 이 숫자를 check digit를 정의하는 데 사용함.
예) $11 - 0 = 11$
Step 4) 전단계에서 구한 check digit가 10이면 X, 11이면 0으로 표시.

1) input_issn_list.txt 파일을 생성하여 다음의 7개의 issn 코드를 저장하시오.

00279358
20493630
0000300X
0057935X
08912350
0891235K
TTTTT



주의 1) Check 기호 “X”처리에 유의하시오. 알파벳 대문자 “X”임.

주의 2) 참고로 입력한 예제 ISSN 코드 7개 검증결과는 다음과 같음. 검증시 활용하시오.

- o 검증결과 맞는 코드 : 00279358, 20493630, 0000300X
- o 검증결과 틀린 코드 : 0057935X, 08912350
- o 형식자체가 잘못된 코드 (숫자와 X이외의 문자가 들어간 경우) : 0891235K, TTTTT

2) 독립된 class로 구현

2-1) 1)단계에서 생성한 input_issn_list.txt에서 issn number를 하나씩 읽어들인다.

2-2) 2-1)단계에서 읽어드린 issn number를 주어진 ISSN 번호검증 알고리즘을 이용하여 다음과 같이 결과를 차례차례 출력한다.

```
>> input_issn_list.txt에서 ISSN 번호를 차례로 읽어옵니다.
>> 파일에서 읽어온 내용 : ['00279358\n', '20493630\n', '0000300X\n', '0057935X\n', '08912350\n', '0891235K\n', 'TTTTT']
>> ISSN code 검증을 시작합니다.

>> 1번째 ISSN code : 00279358
>> ISSN 코드 00279358은(는) 검증되었습니다.

>> 2번째 ISSN code : 20493630
>> ISSN 코드 20493630은(는) 검증되었습니다.

>> 3번째 ISSN code : 0000300X
>> ISSN 코드 0000300X은(는) 검증되었습니다.

>> 4번째 ISSN code : 0057935X
>> ISSN 코드 0057935X은(는) 검증되지 않았습니다.

>> 5번째 ISSN code : 08912350
>> ISSN 코드 08912350은(는) 검증되지 않았습니다.

>> 6번째 ISSN code : 0891235K
>> Exception) 잘못된 코드. 마지막 문자가 X가 아닙니다.

>> 7번째 ISSN code : TTTTT
>> Exception) 잘못된 코드. 숫자나 X이외의 문자를 포함하고 있습니다.
```

2-3) 2-2)단계에서 결과를 출력하면서 최종적으로 다음의 dictionary를 만든다.

```
dic_cert = { "00279358" : "RIGHT", "20493630" : "RIGHT", "0000300X" : "RIGHT",
             "0057935X" : "WRONG", "08912350" : "WRONG", "TTTTT" : "WRONG" }
```

```
>> 생성된 Dictionary입니다 : {'00279358': 'RIGHT', '20493630': 'RIGHT', '0000300X': 'RIGHT', '0057935X': 'WRONG', '08912350': 'WRONG', '0891235K': 'WRONG', 'TTTTT': 'WRONG'}
```

3) 0891235K, TTTTT와 같이 숫자와 X이외의 잘못된 문자가 들어간 경우를 처리하는 user-defined exception handler를 정의하시오. (독립된 class로 구현)

4) 2-3)단계에서 생성한 dic_cert를 pickling을 이용하여 certify_issn_number.pkl에 저장 (dump)한다. (독립된 class로 구현)

5) certify_issn_number.pkl에 저장된 내용을 다시 읽어(load)들여 출력한다. (독립된 class로 구현)

```
>> Pickling을 시작합니다.  
>> Dump : certify_issn_number.pkl 파일을 생성했습니다.  
  
>> Load : certify_issn_number.pkl 파일을 읽어옵니다.  
>> Unpickled 결과입니다 : {'00279358': 'RIGHT', '20493630': 'RIGHT', '0000300X': 'RIGHT', '0057935X': 'WRONG', '08912350': 'WRONG', '0891235K': 'WRONG', 'TTTTT': 'WRONG'}
```

- END -