

Assignment 01

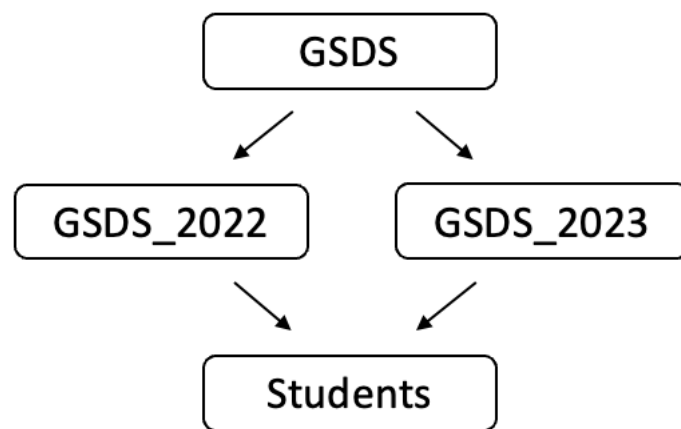
[2023-2] 데이터사이언스를 위한 컴퓨팅의 기초 (M3239.005600 001)

Due: 2023 년 10 월 6 일 금요일 23 시 59 분

1. GSDS Graduate School [30pts]

아래 지시문에 따라 GSDS 대학원 학생의 이수 요건을 확인하는 코드를 완성하시오.

Instruction:



1. GSDS class 는 GSDS 대학원의 석사 및 박사 학생 수와 학생 이름 및 학생이 수강한 과목, 학점, 성적으로 이루어진 학생 정보를 관리한다.
2. GSDS_2022 class 와 GSDS_2023 class 는 각 연도별 입학한 GSDS 학생들의 이수를 위한 요구 학점 및 요구 평균 성적을 나타낸다.
3. Student class 는 각 학생이 수강한 과목, 학점, 성적으로 이루어진 학생 정보를 업데이트하고 이수 현황, 이수 요건을 만족하는 지 확인한다.
4. GSDS, GSDS_2022, GSDS_2023 class 는 수정이 불가하고 오로지 Student class 의 updateSubject, checkSubject, checkGraduation 의 function body 만 입력한다.
5. 재수강은 인정되지 않는다.
6. 프린트문은 수정하지 않는다.

Quiz

Q1.1 updateSubject 함수를 완성하시오.

Hint. updateSubject 함수는 학생 이름과 학생이 수강한 과목명, 학점 수, 성적을 입력 받아 GSDS class 의 students 에 저장한다.

Q1.2 checkSubject 함수를 완성하시오.

Hint. checkSubject 함수는 학생이 수강한 과목들의 총 학점 수와 평균 성적을 출력한다.

Q1.3 checkGraduation 함수를 완성하시오.

Hint. checkGraduation 함수는 GSDS_2022 class 와 GSDS_2023 class 의 정보를 참고하여 학생이 요구 학점 수와 요구 평균 성적을 만족하는 지 확인한다.

Implementation:

```
grad = GSDS()
grad_2022 = GSDS_2022(grad)
grad_2023 = GSDS_2023(grad)
taegyoon = Student(grad_2022, 'taegyoon', 'phd')
jiyoon = Student(grad_2023, 'jiyoon', 'ms')
grad.studentTotal()

The total number of students at GSDS graduate school is 1 ms and 1 phd.

taegyoon.updateSubject("CFDS", 3, 4.3)
taegyoon.updateSubject("CFDS1", 3, 4.0)
taegyoon.updateSubject("CFDS2", 3, 4.0)
taegyoon.updateSubject("MLDL1", 3, 3.3)
taegyoon.updateSubject("MLDL2", 3, 3.3)
taegyoon.updateSubject("MLDL1", 3, 4.3)
taegyoon.checkSubject()
taegyoon.checkGraduation()

studentName : taegyoon, subjectName : CFDS, credit : 3, score : 4.3
studentName : taegyoon, subjectName : CFDS1, credit : 3, score : 4.0
studentName : taegyoon, subjectName : CFDS2, credit : 3, score : 4.0
studentName : taegyoon, subjectName : MLDL1, credit : 3, score : 3.3
studentName : taegyoon, subjectName : MLDL2, credit : 3, score : 3.3
Retakes are not permitted
Total Credit : 15, Avg Grade : 3.8
Graduation requirements are not met.

jiyoon.updateSubject("CFDS", 3, 4.3)
jiyoon.updateSubject("CFDS1", 3, 3.7)
jiyoon.updateSubject("CFDS2", 3, 4.0)
jiyoon.updateSubject("MLDL1", 3, 4.0)
jiyoon.updateSubject("MLDL2", 3, 4.0)
jiyoon.checkSubject()
jiyoon.checkGraduation()

studentName : jiyoon, subjectName : CFDS, credit : 3, score : 4.3
studentName : jiyoon, subjectName : CFDS1, credit : 3, score : 3.7
studentName : jiyoon, subjectName : CFDS2, credit : 3, score : 4.0
studentName : jiyoon, subjectName : MLDL1, credit : 3, score : 4.0
studentName : jiyoon, subjectName : MLDL2, credit : 3, score : 4.0
Total Credit : 15, Avg Grade : 4.0
Graduation requirements are met.
```

2. 전자출결시스템 [30pts]

아래 지시문에 따라 전자출결시스템이 강의별 출석현황을 산출하는 코드를 완성하시오.

Instruction:

1. 컴기대학교에서는 전자출결시스템을 도입하였다. 전자출결시스템은 다음과 같은 방식으로 작동한다.
 - 학생들 한 명 한 명은 강의실입장 및 강의실퇴장 시, 강의실 문에 설치된 전자출결장치에 학생증을 태그한다.
 - 학생별 강의실입장시각, 강의실퇴장시각이 시스템에 기록된다. 이때, 강의실입장시각은 강의시작시간 10 분 전부터 기록된다.
(ex. 강의시작(11:00)일 때, 해당 강의에 대한 강의실입장시각은 10:50 부터이다)
 - 수업 종료로부터 3 시간 뒤에, 강의수강을 위해 강의실에 출입했던 모든 학생들의 출입시각기록이 class_students_data0.txt 로 저장된다.
 - 강의 정보인 'class_info0.txt' 파일과 이전 단계에서 저장된 'class_students_data0.txt' 파일은 '출석현황 산출' 알고리즘을 거쳐 해당 강의의 출석현황 결과로 출력된다.
(ex. Machine_Learning 과목의 출결현황입니다. 총수강생: 15 명 中 출석: 9 명, 지각 및 조퇴: 4 명, 결석: 2 명)
2. '출석현황 산출' 알고리즘이 작동하는 기본 논리는 다음과 같다.
 - 출석 : ㉠와 ㉡ 모두 충족 시
 - ㉠강의시간(강의종료시각 - 강의시작시각)의 90%이상 시간만큼 강의 수강
 - ㉡강의시작시각으로부터 10 분 뒤까지 강의실 입장
(ex. 강의시작(11:00)일 때, 11:10 까지 강의실 입장 필요)
 - 지각 및 조퇴: ㉢ 또는 ㉣
 - ㉢강의시작시각 이후 10 분이 넘어서 교실에 입장한 경우
 - ㉣강의시간의 50%초과 90%미만만큼 강의수강
 - 결석 : ㉤ 또는 ㉦
 - ㉤강의시간의 50% 이하의 시간만큼 강의 수강
 - ㉦아예 강의실에 출입하지 않은 경우(강의실 출입시각 기록이 없음)

※ 만약 '지각 및 조퇴'와 '결석' 요건을 모두 만족하는 경우, '결석'임.
(ex. 강의시작시간 10 분 이후에 입장과 강의시간의 50%이하만큼 강의 수강 시 → '결석')
3. 유의사항
 - 강의시작시각, 강의종료시각, 강의실입장시각, 강의실퇴장시각은 'HH:MM' 형태로 기록하며 범위는 '00:00 ~ 23:59'임. (분 단위까지만 기록됨. 초 단위는 고려하지 않음) 모든 강의의 강의시작시각과 강의종료시각은 항상 같은 날임.

- 학생의 강의실퇴장시각이 강의종료시각 이후일 경우, 강의종료시각을 기준으로 강의수강시간을 산정함.
- 수강생들의 강의실 출입기록에는 반드시 강의실입장시각과 강의실퇴장시각이 '한 쌍' 존재한다.
(존재하지 않는 기록: 강의실입장시각만 있는 경우, 강의실퇴장시각만 있는 경우, 한 학생이 강의실입장과 퇴장을 두 번 이상씩 하는 경우)
- 'class_info0.txt'와 'class_students_data0.txt'의 형식과 예시는 다음과 같다.

<ul style="list-style-type: none"> • class_info0.txt (형식) 과목명 강의시작시각 강의종료시각 총수강생 	<ul style="list-style-type: none"> • class_students_data0.txt (형식) 강의실출입시각 강의실퇴장시각 ← 이 형태로 여러 줄
<div>▼예시▼</div> <div>Computer_Science 11:00 12:30 30</div>	<div>▼예시▼</div> <div>10:52 12:30 10:57 12:25 11:05 12:32 ...</div>

- 프린트문은 수정하지 않는다.

Quiz

Q2.1 main 함수를 완성하시오.

Hint : string index 를 적절히 활용한다.

Q2.2 ~ Q2.3 attendance_record 함수를 완성하시오.

Hint1 : datetime.timedelta() method 를 적절히 활용한다.

Hint2 : 조건문 구현 시, 조건문의 처리순서에 유의한다

.

Implementation:

```
# class_info1, class_students_data1
class_info_file = 'class_info1.txt'
class_students_data_file = 'class_students_data1.txt'
main(class_info_file, class_students_data_file)

Computer_Science 과목의 출결현황입니다. 총수강생: 30명 중 출석: 19명, 지각 및 조퇴: 7명, 결석: 4명

# class_info2, class_students_data2
class_info_file = 'class_info2.txt'
class_students_data_file = 'class_students_data2.txt'
main(class_info_file, class_students_data_file)

Machine_Learning 과목의 출결현황입니다. 총수강생: 15명 중 출석: 9명, 지각 및 조퇴: 4명, 결석: 2명

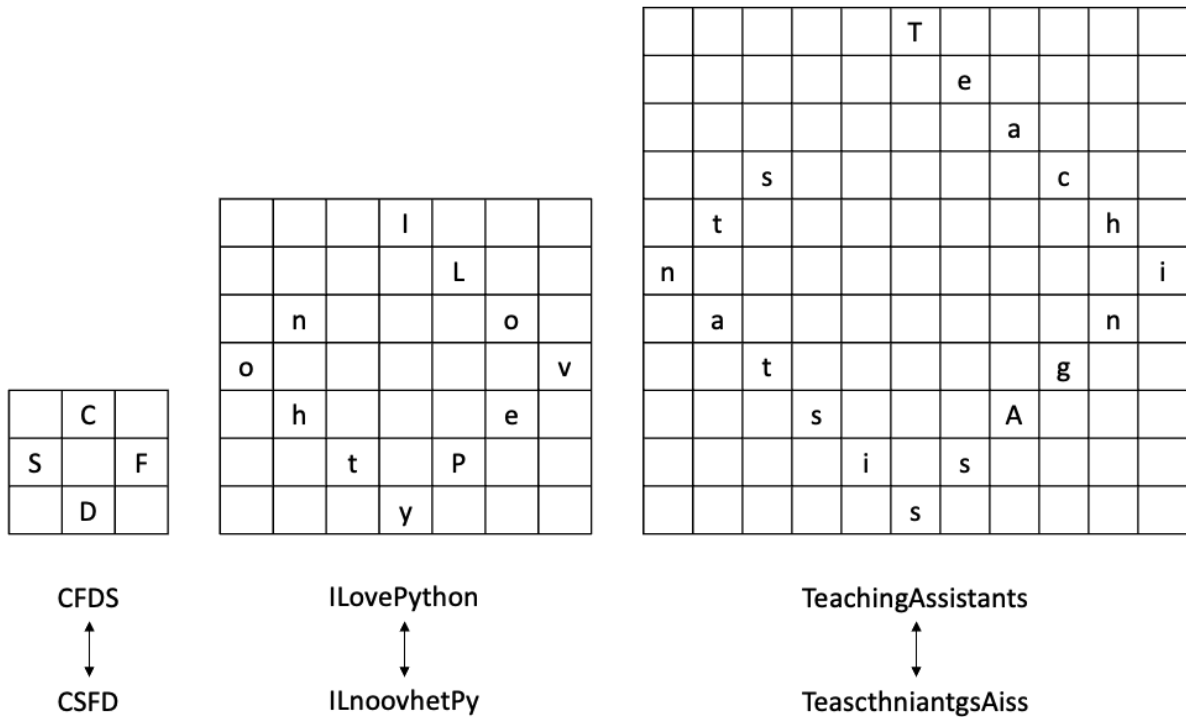
# class_info3, class_students_data3
class_info_file = 'class_info3.txt'
class_students_data_file = 'class_students_data3.txt'
main(class_info_file, class_students_data_file)

Data_Science_Seminar 과목의 출결현황입니다. 총수강생: 51명 중 출석: 34명, 지각 및 조퇴: 11명, 결석: 6명
```

3. Encryption & Decryption [20pts]

아래 지시문에 따라 문장을 암호화, 복호화하는 코드를 완성하시오.

Instruction:



1. 문장을 입력 받으면 정사각 테이블에서 12 시 방향을 시작으로 3 시, 6 시 9 시 방향 순서로 대각선 이동을 하면서 단어를 나열하는데 이 때 정사각 테이블의 길이는 홀수로 하고 문장 전체를 표현할 수 있는 크기 중 가장 작게 만든다.
2. 이후 테이블의 위부터 좌에서 우로 읽어가며 문장 순서를 바꿔 암호화를 한다.
3. 암호화한 문장을 다시 복호화를 한다.
4. 프린트문은 수정하지 않는다.

Quiz

Q3.1 encryption 함수를 완성하시오. [10pts]

Hint. 먼저 테이블의 길이부터 구하고 각 알파벳이 몇번째 행에 해당하는 지 저장한다. 행 단위로 입력하되 같은 행에 해당하는 알파벳 순서에 유의한다.

Q3.2 decryption 함수를 완성하시오. [10pts]

Hint. 먼저 테이블의 길이부터 구하고 알파벳이 하나만 들어가는 행을 구분한다.
나머지는 암호화된 문장에서 알파벳 2 개씩 1 행에 해당한다는 것에 유의한다.

Implementation:

```
crypto = encryption("CFDS")
decryption(crypto)

CSFD
CFDS

crypto = encryption("ILovePython")
decryption(crypto)

ILnoovhetPy
ILovePython

crypto = encryption("TeachingAssistants")
decryption(crypto)

TeascthniantgsAiss
TeachingAssistants
```

Cautions

1. 코드 채점 시에는, 예시로 주어진 test case 외에 추가적인 test case 들을 활용하여 채점할 예정으로 edge case 들에 대한 검증을 꼼꼼히 하고 제출하기 바랍니다.
2. 작성한 ipynb 파일을 Assignment1_학생이름_학번.zip 로 압축하여 제출하기 바랍니다. (아래 예시 참고)

> Assignment1_이순신_2023-12345.zip

이름

Q1.ipynb

Q2.ipynb

Q3.ipynb

3. 제출 양식 미 준수 시 채점 간 오류가 발생할 수 있습니다. 제출 양식 미 준수로 인한 오류 교정은 없습니다. 수강생 분들은 제출 양식을 반드시 확인 바랍니다.