

Pre-URP 2주차 과제 설명

0. 들어가기에 앞서

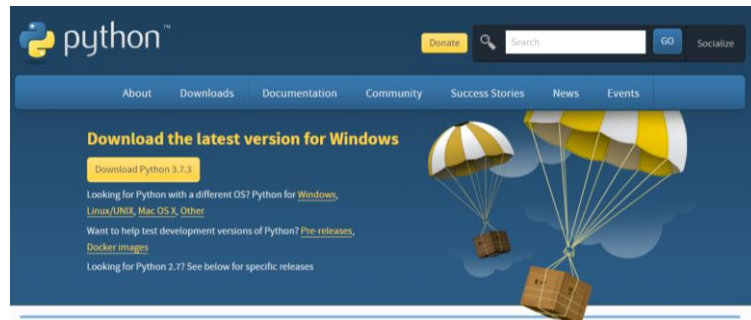
2주차 과제는 총 2개입니다. 마감일은 7월 8일까지입니다. 과제가 어렵거나, 모르는 부분이 있다면, 카카오톡 톡방 혹은 문자, 이메일 coder@dgist.ac.kr로 연락 주시면 상세히 대답해 드리겠습니다.

1. CUBE 프로그래밍 언어 실행시켜 보기 & 기본적인 기능 사용하기

Step A. 파이썬 설치 & 개발환경 구축

(이미 컴퓨터에 파이썬이 설치되어 있을 경우, 이 과정을 건너뛰어도 괜찮습니다.)

현재 CUBE 프로그래밍 언어를 실행하기 위해서는 Python이 필요합니다. 설치를 위해서는 Python 공식 홈페이지(<https://www.python.org/>)에 접속합니다. 그 뒤, Download 버튼을 클릭하고 Download Python 3.7.3을 눌러 프로그램 설치 파일을 받습니다.



다운로드 받은 실행파일을 실행해 Python을 설치합니다. Python이 제대로 설치되어 있는지 알기 위해서, Windows 버튼을 누르고, "실행" 또는 "cmd"를 검색해 실행 창을 킵니다.

```
명령 프롬프트 - python
Microsoft Windows [Version 10.0.17134.829]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\coder>python
Python 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32

Warning:
This Python interpreter is in a conda environment, but the environment has
not been activated. Libraries may fail to load. To activate this environment
please see https://conda.io/activation

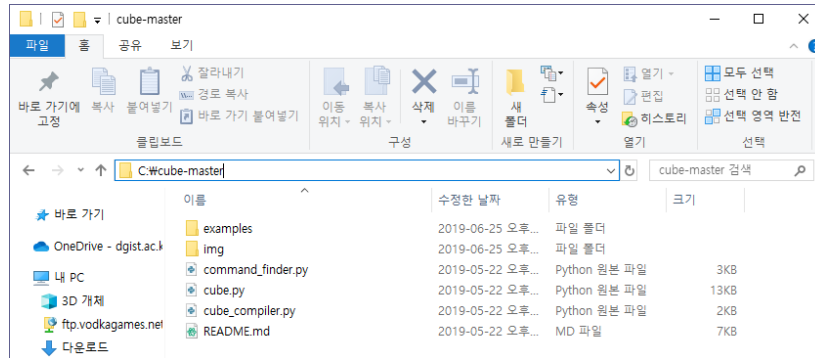
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

위 그림처럼 나오면 성공입니다. 종료를 위해서는 창을 닫거나, exit()을 콘솔창에 입력합니다.

Step B. CUBE git 파일 다운로드 받고 실행하기

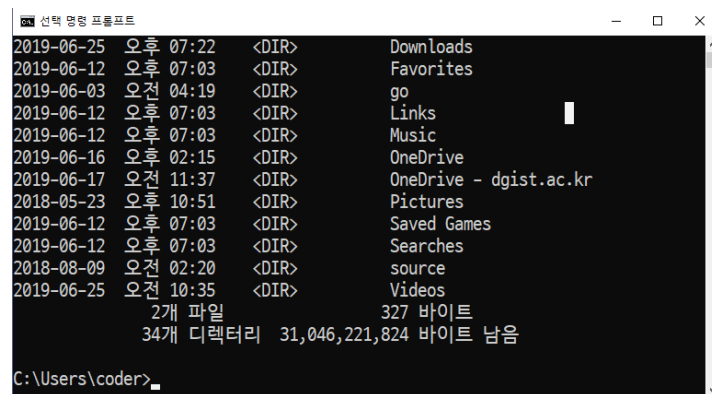
이제 CUBE를 받아 봅시다. CUBE Github 페이지로 갑니다(<https://github.com/r3coder/cube>). 오른쪽에 있는 초록색 "Clone or Download"를 클릭하고, Download ZIP을 눌러서 파일을 받습니다.

받은 파일은 작업하고 있는 폴더가 있다면 그 곳에, 잘 모르겠으면 C 드라이브 바로 아래에 풀어줍니다.

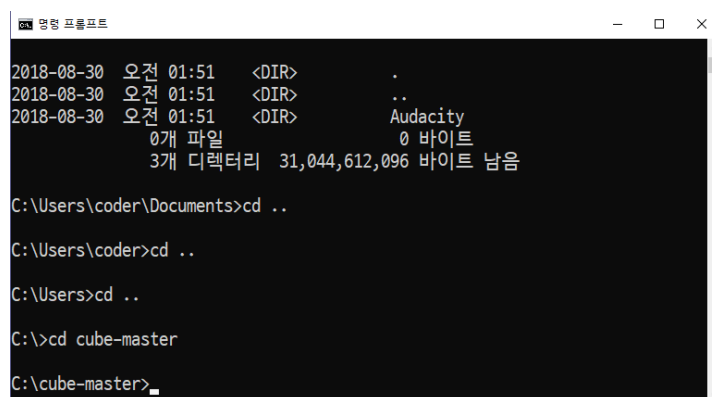


이제, 다시 "실행" 혹은 "cmd"로 콘솔 창을 다시 엽니다. 콘솔 창을 열면, "C:\Users\coder>" 옆에 글자가 깜박일 것입니다. 이 파일 경로는 현재 자신이 있는 위치를 의미합니다.

"dir"을 입력해 현재 폴더에 무엇이 있는지 확인합니다. dir은 Directory의 약자입니다.



아까 압축은 C 드라이브 바로 아래에 풀었으므로, "cd .."을 두 번 입력해 위치를 이동합니다. cd는 Change Directory의 약자입니다. 그 뒤, "cd cube-master"를 입력해 해당 폴더로 들어갑니다.



이제, 큐브 프로그램을 실행시켜 볼 차례입니다. 예시로, "python cube_compiler.py -s

examples/hello_world.cube”를 명령창에 쳐 봅시다.

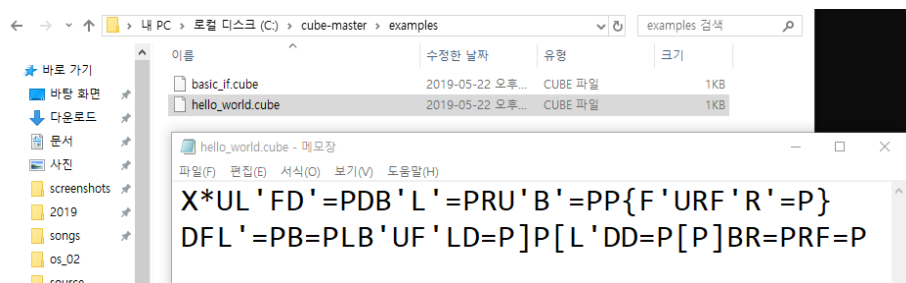
```
C:\cube-master>python cube_compiler.py -s examples/hello_world.cube
Reading From File: examples/hello_world.cube
Script Loaded with Length 77
X*UL'FD'=PDB'L'='PRU'B'='PP{F'URF'R'='P}DFL'='PB=PLB'UF'LD=P]P[L'DD=P[P]BR=PRF=P

*****
Result
*****
Hello, world!

C:\cube-master>
```

결과에 Hello, world!가 나왔습니다.

방금 입력한 `python cube_compiler.py -s examples/hello_world.cube`에 대해서 간단하게 설명하겠습니다. 방금 입력한 코드는, 명령창에 python이라는 프로그램을 사용해서, `cube_compiler.py`라는 파이썬 코드를 실행한다는 의미입니다. 그리고 그 뒤에 따라오는 `-s`는 프로그램 내부에서 스크립트 입력을 받을 것이라고 알려주는 것이고, 그 뒤에 오는 `examples/hello_world.cube`는 입력할 스크립트 파일입니다. 만약 맨 뒤를 다른 파일로 바꿔 주면, 다른 파일을 실행하게 됩니다. “.cube” 확장자는 일반 텍스트 에디터(메모장) 등으로 열어서 수정할 수 있습니다.



메모장으로 hello_world.cube를 연 모습

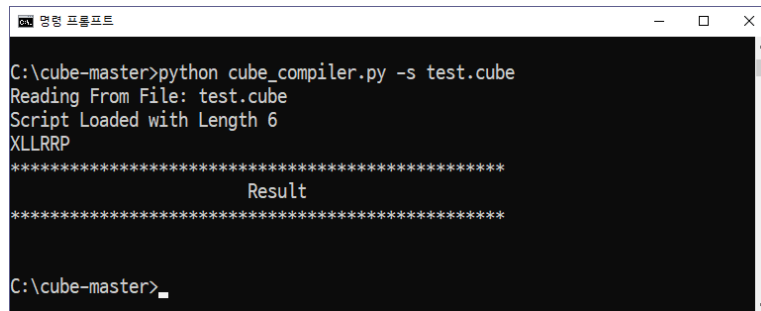
Step C. CUBE 소스 파일 작성하기

이제, 큐브 소스 파일을 만들어 봅시다. 오른쪽 클릭 – 새로 만들기 – 텍스트 파일을 선택한 뒤, 이름을 “test.cube” 혹은 원하는 것으로 바꿉니다.

__pycache__	2019-06-25 오후...	파일 폴더	
examples	2019-06-25 오후...	파일 폴더	
img	2019-06-25 오후...	파일 폴더	
command_finder.py	2019-05-22 오후...	Python 원본 파일	3KB
cube.py	2019-06-25 오후...	Python 원본 파일	13KB
cube_compiler.py	2019-05-22 오후...	Python 원본 파일	2KB
README.md	2019-05-22 오후...	MD 파일	7KB
test.cube	2019-06-25 오후...	CUBE 파일	0KB

방금 만든 test.cube를 오른쪽 클릭한 뒤, 연결 프로그램을 선택하고 메모장을 선택하거나, 자신이 사용하고 있는 텍스트 편집기를 선택합니다. 그런 뒤, CUBE 코드를 작성한 뒤에 저장합니다.

이제, 콘솔 창으로 돌아와서, 방금 만든 코드를 실행해 봅시다. 실행은 "python cube_compiler.py -s <파일 경로>"로 합니다.



Step D. CUBE 배우기

CUBE 언어에 대한 설명은 CUBE 공식 페이지 (<https://github.com/r3coder/cube>)에서 읽을 수 있습니다. 설명이 조금 난해할 수 있으니, 잘 이해가 안 가는 부분에 대해서는 적극적으로 물어봐 주시기 바랍니다.

Step E. 과제

HW 1-1. 숫자 하나를 입력 받고, 해당 숫자를 그대로 출력하는 CUBE 코드를 작성하세요.

힌트: 프로그램 코드는 입력 받기, 반대편 면으로 옮기기, 출력하기 세 과정이 필요합니다.

힌트: 일반적인 루빅스 큐브에서, 한 면에 있는 색을 모두 반대편 면으로 옮긴다고 생각하고 만들어 보세요. (예) 노란색 면과 하얀색 면이 서로 반대 방향에 있을 때, 노란색 면에 중앙을 제외하고 모두 하얀색이 되도록

HW 1-2. 숫자 하나를 입력 받고, 해당 숫자의 2's compliment를 출력하는 CUBE 코드를 작성하세요.

힌트: 아까는 반대편 면으로 옮겨야 한다면, 이번에는 바로 옆 면으로 옮기는 과정이 필요합니다. 아까와 다르게, 입력 받기, 옆 면으로 옮기기, 연산 한 뒤 불러오기, 다시 옆 면으로 옮기기, 출력하기까지 필요합니다.

힌트: 2's compliment에 대한 설명: https://ko.wikipedia.org/wiki/2의_보수

2. 원형 큐 구현하기

Step A. 원형 큐에 대해서

원형 큐는 자료구조의 한 종류입니다. 여기에 새로 설명하는 것보다, 읽어보면 좋을 거 같은 글과 자료를 남기겠습니다.

[자료구조] 큐(Queue)-1 <https://hannom.tistory.com/24>

별첨한 Queue 설명 자료

Step B. 스켈레톤 코드와 과제 설명

본 과제에서는 주어진 스켈레톤 코드(이미 만들어진 코드)에서 함수 안에 내용을 작성하면 됩니다. 미리 만들어져 있는 코드인 CircularQueue.cpp에 대해서 설명하겠습니다.

Line	내용
2~5	필요한 헤더를 include하는 구문입니다. 만약 Visual Studio를 사용하는 경우, 여기에 stdafx.h를 추가할 필요가 있을 수도 있습니다.
8	CircularQueue Class의 선언입니다.
9~14	내장 변수들에 대한 설명입니다. 더욱 자세한 설명은 아래에 하겠습니다.
16~21	CircularQueue Class의 생성자입니다. capacity를 입력 받아, 해당 공간만큼 정보를 담을 수 있습니다.
24~36	직접 작성해야 하는 Add 함수입니다. 만약 Queue가 가득 찼을 경우의 예외처리는 되어 있습니다.
38~50	직접 작성해야 하는 Pop 함수입니다. 만약 Queue가 비었을 경우의 예외처리는 되어 있습니다.
52~54	다음에 Pop되는 원소를 출력하는 Tail 함수입니다.
60~92	입력을 받고 CircularQueue에 직접 넣어주는 구문들입니다.

내장 변수 설명

int this->capacity: 원형 큐의 최대 수용량입니다.

int this->size: 현재 들어가 있는 데이터의 수입니다.

int* this->data: 데이터를 담고 있는 배열입니다.

int* this->head: 큐에 데이터가 들어갈 장소를 가리키는 포인터입니다.

int* this->tail: 큐에서 데이터가 빠지는 장소를 가리키는 포인터입니다..

작성해야 하는 코드는 다음과 같습니다.

Add(int value): 큐에 value 값을 추가합니다. 더 이상 값을 추가할 수 없을 때의 예외 처리는

되어 있습니다.

Pop(): 큐에서 데이터 하나를 빼고, 그 값을 반환합니다. 더 이상 값을 뺄 수 없을 때의 예외 처리는 되어 있습니다.

Step C. 첨부 파일 설명

첨부 파일에 대해서 설명합니다.

CircularQueue.cpp: 스켈레톤 소스 코드입니다.

input.txt: 입력 예시 파일입니다.

output_answer.txt: 출력 예시 파일입니다. 위 input.txt를 입력으로 받으면, 출력이 나와야 합니다.

Makefile: 채점용 검사 파일입니다.

Step D. 보고서

2번 과제에는 소스 코드와 보고서를 동봉해야 합니다. 보고서에는 다음과 같은 내용이 들어가야 하며, 최대 A4 1장 정도로 작성하면 됩니다.

- 이름
- 코드 설명

3. 결과 제출

과제 제출은 아래 파일들을 이름_HW2.zip 혹은 이름_HW2.tar.gz로 압축해 과제제출 게시판에 게시합니다.

Prob1-1.cube – 과제 1-1번의 큐브 소스 코드

Prob1-2.cube – 과제 1-2번의 큐브 소스 코드

CircularQueue.cpp – 과제 2번의 cpp 소스 코드

Report.docx / Report.pdf – 과제 2번의 보고서 파일(MS Word 혹은 pdf)