

CSED232 Object-Oriented Programming (Spring 2020)

Programming Assignment #1

-Programming Practice-

Due data: 2020년 4월 3일 금요일 23:59

담당교: 남석현 (obiwan96@postech.ac.kr)

첫번째 assignment는 학생들이 C프로그래밍을 배웠던 것을 remind하고 C++ 프로그래밍의 기초적인 실습, 그리고 컴파일 환경 등을 세팅해서 컴파일 해 볼 수 있게 하는게 목적입니다.

채점 방식

문제별로 배포된 예시 dataset을 포함한 채점용 dataset 15개를 모두 통과하면 총점을 받을 수 있다. 채점용 dataset에는 예시 dataset에 없는 data도 존재하니 본인이 test case를 직접 만들어 테스트해볼 것을 권장함.

전체 dataset을 통과하지 못하여도 정답을 맞춘 test case에 대해 부분 점수를 인정한다.

모든 test case에 대하여 3초 내에 결과가 출력되어야 한다. (초과시 0점)

모든 input과 output은 STDIN과 STDOUT을 사용한다. (파일 입출력 X)

각 문항별로 적힌 배점에 보고서 점수 10점을 더해 총 100점 만점으로 한다.

주의사항

동적 메모리 할당 사용시, 할당 해제를 제대로 하지 않을 경우 해당 문제 10%감점.

Compile이 안되거나 실행이 안될 경우 0점 (minGW상에서 g++ compiler 사용).

보고서와 각 코드 main 위에 (주석으로) Honor code를 명시해야 함 (없을 시 0점).

Cheating은 0점, 참고한 자료 및 사이트는 보고서에 reference로 적는다. 모든 코드는 인터넷상의 코드와 대조해볼 것이니 참조할 때 이해한 후 본인이 새로 짤 것.

문제에 따라 100만 크기의 array를 사용해야 하는데, local variable로는 해당 크기의 array가 잡히지 않는다. 본 과제에 대해서만 global variable의 사용을 권장함.

일반적인 프로그래밍의 경우에는 global variable 사용이 권장되지는 않으나, 본 과제에 대한 프로그래밍의 경우 global variable 없이 문제를 푸는 것이 어려울 수 있음.

출력 양식을 지킬 것. 예를 들어 '3'이 답인 경우, '3' 이외에 'Wn'과 같은 부가적인 whitespace, 캐리지 리턴 등의 문자를 포함할 경우 0점 처리됨.

제출 방식

제출 코드는 문항별로 'prob_1', 'prob_2', 'prob_3' 폴더를 만든다. 각 폴더에는 코드 파일들(*.cpp, *.h)을 저장한다. 코드에는 간단한 주석을 통해 설명을 포함시킨다. 상위 폴더에 반드시

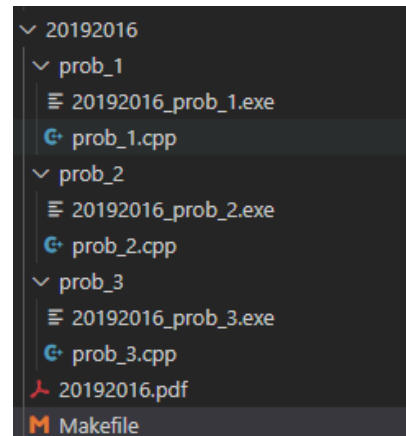
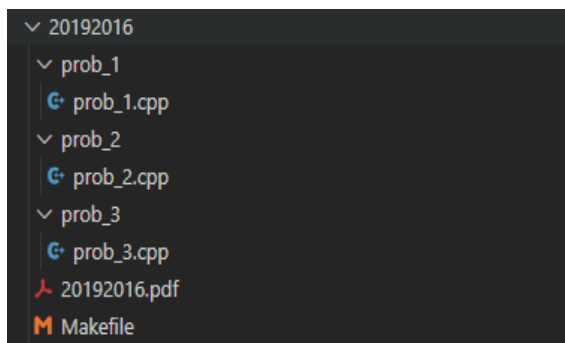
Makefile을 만들어서 make all를 이용하여 세 문제 모두 컴파일하여 각 문제 폴더에 실행파일이 생성될 수 있어야 한다. make all를 통해 생성되는 실행파일 이름은 학번_prob_문제번호.exe 로 통일하도록 한다. 이 때, 소스 코드 파일 외에, 불필요한 파일들(*.obj, *.o, *.exe, a.out 등)은 zip 파일에 포함하지 않는다.

제출 보고서는 보고서 양식에 따라 그림 포함 5페이지 이내로 작성하며, PDF 파일로 제출한다. 파일 이름은 '학번.pdf' 로 한다.

코드와 보고서 파일 전체를 '학번.zip' 파일로 묶어서 제출한다.

제출 방식을 지키지 않을 경우 총점에서 10% 감점

아래 왼쪽 사진은 제출 파일 예시이며 오른쪽은 해당 폴더에서 make all을 한 후의 모습이다.



채점 기준

- 프로그램 기능 (전체점수의 50%)
프로그램이 요구 사항을 모두 만족하면서 올바르게 실행되는가?
조교가 프로그램을 실행해 보고 채점
- 프로그램 설계 및 구현 (전체점수의 35%)
요구 사항을 만족하기 위한 변수 및 알고리즘 설계가 잘 되었는가?
설계된 내용이 요구된 언어를 이용하여 적절히 구현되었는가?
보고서에 설명된 프로그램 설계 및 구현에 관한 내용을 기준으로 채점
보고서에 설명된 내용을 프로그램의 소스 코드를 참조하여 확인
- 프로그램의 가독성(readability) (전체점수의 5%)
프로그램이 읽기 쉽고 이해하기 쉽게 작성되었는가?
변수 명이 무엇을 의미하는지 이해하기 쉬운가?
프로그램의 소스 코드를 이해하기 쉽도록 주석을 잘 붙였는가?
- 보고서 구성 및 내용, 양식 (전체점수의 10%)
보고서는 적절한 내용으로 이해하기 쉽고 보기 좋게 잘 작성되었는가?
보고서의 기술적인 내용과 구성이 적절한가, 설명이 간결하고 명료하게 되었는지를 기준으로 채점
보고서의 양식을 잘 따랐는가?
- 추가 점수 (최대 전체점수의 10%)
요구 사항 외의 추가 기능 또는 독창적으로 생각한 기능을 구현한 경우

추가된 기능의 독창성과 난이도를 기준으로 채점

□ 감점

제출 기한이 지나면 얻은 총점의 20% 감점

추가로 하루(24시간) 늦을 때마다 20%씩 감점

1일 이내 지연: 20% 감점, 2일 이내 지연: 40% 감점, 5일 이상 지연: 0점

다른 사람의 프로그램이나 인터넷에 있는 프로그램을 복사(copy)하거나 간단히 수정해서 제출하면 학점은 무조건 'F'가 된다. 이러한 부정행위가 발견되면 학과에서 정한 기준에 따라 추가의 불이익이 있을 수 있다.

Makefile이 없거나 make를 통해 정상적으로 컴파일이 되지 않을 경우 모든 test data에 대하여 실행되지 않는 것으로 본다.

[참고] C++에서 C의 Input, Output 사용

cstdio 파일을 include한다.

printf -> std::printf, scanf -> std::scanf로 변환하여 사용한다

예시

```
#include <cstdio>

int main()
{
    int var;

    std::scanf("%d",&var);
    std::printf("Hello World!\n");
    std::printf("%d\n",var);

    return 0;
}
```

위 프로그램은 int형 변수를 1개 입력받고, "Hello World!"문구와 입력 받은 변수를 출력한다.

```
123
Hello World!
123
```

기타

C++를 알고 있는 학생들은 프로그램을 C++로 작성해도 좋습니다. C++를 아직 모르는 학생들은 C를 사용하면 됩니다.

이미 STL을 알고 있는 경우에는 사용 가능하고 (e.g., std::queue, std::vector), 모르는 경우에는 그냥 array를 사용하면 됩니다.

잘못된 경우의 입력은 주어지지 않습니다. 예외처리를 할 필요는 없습니다.

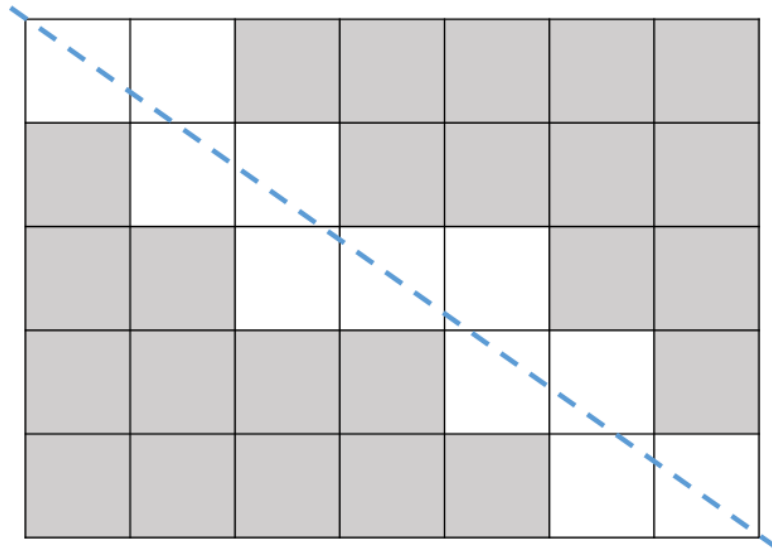
[참고] Makefile

Makefile 만드는 법에 대해선 구글링을 조금만 해도 쉽게 알 수 있기 때문에 따로 설명을 만들지 않았습니다. 다만 본 과제는 윈도우에서 컴파일하는 것을 기본으로 하고 있기 때문에 clean 명령은 보통의 설명과 달리 rm 대신 del 명령어를 사용해야 합니다. 그 외에는 똑같이 하면 됩니다.

1. 온전한 사각형 (배점: 20점)

난이도 : 하, 키워드 : 최대공약수

재영에게는 가로 길이가 W , 세로 길이가 H 인 모눈종이가 있다. 그런데 재영이 잠시 화장실 청소를 하러 간 사이, 재영이 키우는 고양이인 두부가 모눈종이를 정확히 대각선모양으로 잘라버렸다. 잘린 모눈종이를 발견한 재영은 모눈종이의 일부 사각형들은 잘려 사용할 수 없음을 깨달았다. 잘리지 않은 온전한 사각형은 과연 몇 개일까?



예를 들어 위는 가로가 7, 세로가 5인 모눈종이이다. 재영이 온전한 사각형만 색칠을 해보니 24 개임을 알 수 있었다. W 와 H 가 입력될 때 온전한 사각형의 수를 출력하시오.

[입력]

W 와 H 가 공백을 간격으로 입력된다. ($1 \leq W, H \leq 10^8$) 단, 곱해서 10^9 을 넘는 값은 입력되지 않는다.

[출력]

온전한 사각형의 수를 출력한다.

[입출력 예시]

Input	Output
7 5	24

2. 리코더 (배점: 30점)

난이도 : 중, 키워드 : 탐욕알고리즘 (Greedy Algorithm)

동우네 반이 점심을 먹으러 간 사이, 두부가 몰래 들어와 동우네 반 물건들을 훔쳤다! 다른 물건들은 천천히 찾아도 되지만, 당장 다음 수업이 음악 수업이라 리코더가 필요하다. 다행히 동우네 반은 두부의 침입에 익숙해져 몇몇 학생들은 여분의 리코더를 가지고 있다고 한다. 그런데 동우네 반 학생들은 청결을 중요시하기 때문에 본인과 친한, 바로 앞 번호 학생과 바로 뒤 번호 학생에게만 리코더를 빌려줄 수 있다고 한다. 이에 동우는 가장 많은 학생들이 음악 수업에 리코더를 가져갈 수 있도록 방법을 찾아야 한다. 전체 학생 중 리코더를 잃어버린 학생의 번호와 여분의 리코더를 가지고 있는 학생의 번호가 주어지면 리코더를 들고 음악 수업에 갈 수 있는 최대 학생 수를 구하시오.

단, 학생의 번호는 1번부터 학생 수인 n 번까지로 부여되어 있으며, 여분의 리코더는 하나밖에 없어 여분의 리코더를 가지고 있는 학생이 도난을 당한 경우에는 본인은 쓸 수 있지만 빌려주지는 못한다.

예를 들어, 5명의 학생 중 2번 학생과 4번 학생이 리코더를 잃어버렸고 3번학생이 여분의 리코더를 가지고 있다면 3번 학생이 2번학생에게 리코더를 빌려주어 4명의 학생이 리코더를 가져갈 수 있다.

[입력]

첫번째 줄에 동우네 반 전체 학생수와 도난당한 학생 수, 여분 리코더를 가지고 있는 학생 수가 주어진다. ($0 \leq n \leq 10^5$, $0 \leq k, l \leq n$)

둘째 줄에는 k 개의 리코더를 도난 당한 학생들의 번호가 공백을 간격으로 입력된다.

셋째 줄에는 l 개의 여분의 리코더를 가지고 있는 학생들의 번호가 공백을 간격으로 입력된다.

[출력]

리코더를 가져갈 수 있는 최대 학생 수를 출력시킨다.

[입출력 예시]

Input	Output
5 2 1 2 4 3	4
Input	Output
8 3 2 1 3 5 4 5	7

3. 위스키 시음 (배점: 40점)

난이도 : 상, 키워드 : 동적 계획법 (Dynamic Programming)

두부는 애주가이다. 그런데 집 앞에서 위스키 시음회가 열린다고 해서 참가하러 갔다. 위스키 시음회에는 테이블 위에 다양한 위스키가 들어있는 잔이 일렬로 놓여있었다. 두부는 최대한 많은 위스키를 맛보려 했는데, 위스키 시음회에는 다음과 같은 엄격한 규칙이 있다.

1. 위스키 잔을 선택하면 그 잔에 있는 위스키는 모두 마셔야 하며 마신 후에는 제자리에 다시 뒹야 한다.
2. 연속으로 이어서 세 잔을 마실 수 없다.

두부를 도와 두부가 최대한 많은 양의 위스키를 마실 수 있도록 하자. 1부터 n 까지의 번호가 붙어 있는 위스키 잔이 순서대로 테이블 위에 있으며 각 위스키의 양이 주어졌을 때 두부가 마실 수 있는 최대의 양을 출력하시오.

[입력]

첫번째 줄에 전체 위스키 잔의 수가 주어진다. ($1 \leq n \leq 10,000$)

둘째 줄부터 $n+1$ 번째 줄까지 위스키 잔에 들어있는 위스키의 양이 순서대로 주어진다. 각 위스키의 양은 1000이하의 자연수이다.

[출력]

두부가 마실 수 있는 위스키의 최대량을 출력한다.

[입출력 예시]

Input	Output
6 6 10 13 9 8 1	33