

프로그래밍 연습

Project #2

“XOR” Contest

Project Description

- 재호는 흑백화면에 그림을 그리는 모바일 앱을 만들려고 한다.
- 그러나 이 모바일 기기는 특정 사각형 안에 있는 픽셀들의 색을 바꾸는 "XOR 작업"만이 가능하다.
- 재호가 원하는 그림을 최소한의 XOR 작업으로 그릴 수 있도록 도와주자.



Project Description

- 화면은 가로 N개, 세로 N개의 픽셀로 표현된다.
- 화면 왼쪽 맨 위의 좌표는 (1, 1)이고 row는 아래로 갈수록, col은 오른쪽으로 갈수록 증가한다.
- 0은 흰색을 1은 검정색을 나타낸다.
- 하나의 XOR 작업 $XOR(row1, col1, row2, col2)$ 는 row가 $[row1, row2]$ 에 속하고 col이 $[col1, col2]$ 에 속하는 사각형 영역의 색을 바꾼다.
 - 0인 픽셀은 1로, 1인 픽셀은 0으로

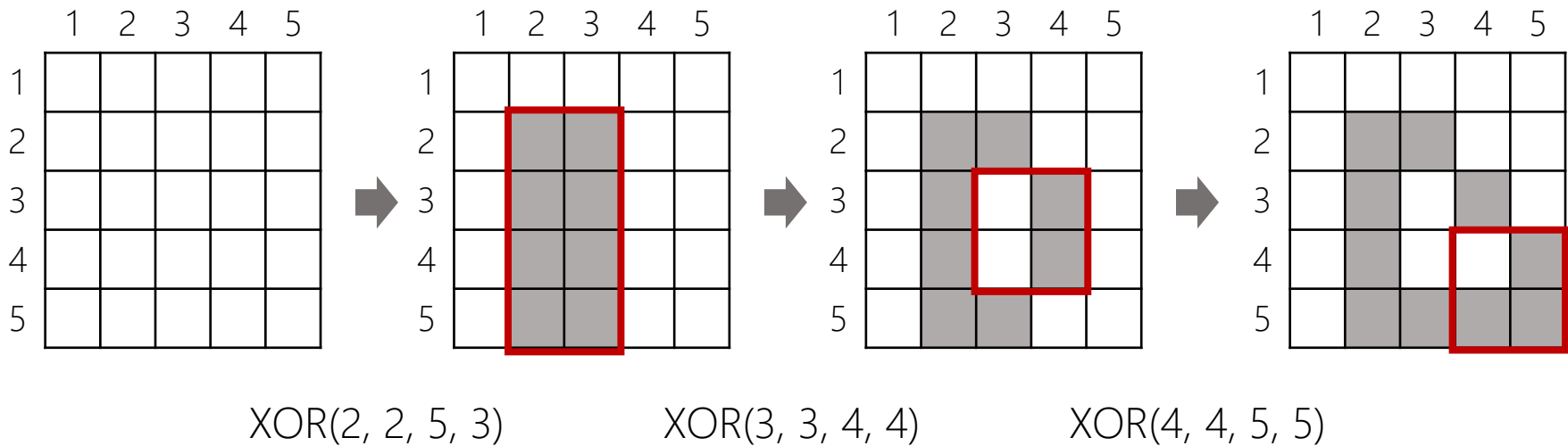
	1	2	3	4	5
1					
2					
3					
4					
5					

$XOR(2, 2, 5, 3)$

	1	2	3	4	5
1					
2					
3					
4					
5					

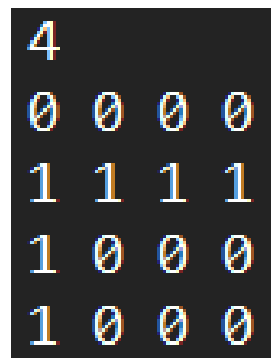
Project Description

- 초기 화면은 모두 흰색(0)으로 칠해져 있다.



Project Description

- 입출력은 stdin, stdout을 통해 이루어진다(scanf, printf).
- 입력에는 재호가 원하는 그림이 주어진다.
- 입력의 첫 줄에는 그림의 높이와 너비를 나타내는 N ($1 \leq N \leq 500$)이 주어지고 이후 N 줄에 걸쳐 N 개의 픽셀이 주어진다.



```
4
0 0 0 0
1 1 1 1
1 0 0 0
1 0 0 0
```

Project Description

- 출력의 첫 줄에는 필요한 XOR 작업의 개수 K를 출력한다.
- 이후 K 줄에는 각 XOR 작업에 대해 (row1, col1, row2, col2) 순으로 출력한다. 줄 끝에는 공백이 없다.
 - XOR 작업 사이에는 교환법칙이 성립하므로 출력 순서는 상관없다.

<입력>

4				
0	0	0	0	
1	1	1	1	
1	0	0	0	
1	0	0	0	

<출력>

2				
2	1	4	4	
3	2	4	4	

Submission

- 작성한 c 소스파일과 Makefile을 함께 [prgxx].zip 파일로 압축하여 moodle의 "Project 2" 항목에 업로드 한다. (파일명 포맷이 맞지 않을 경우 감점됨.)
 - ex) prg01.zip, ~~prg1.zip~~
- 필요한 경우, 파일을 여러 개로 분할하여 프로그램을 작성 후 제출해도 된다. (헤더파일을 만들었을 경우, 함께 제출)
- 채점은 다음과 같이 진행된다.

압축을 해제한 뒤, 해당 폴더에서

make clean

make all

을 순서대로 입력했을 때 해당 폴더에서 xor이라는 실행파일이 만들어지면 채점을 시작.

만약 위와 같이 진행했는데 xor이라는 실행파일이 만들어지지 않으면 채점을 멈추고 감점과 함께 재제출을 요구하는 공지를 할 예정

Submission

- 제출 기한은 12월 21일 (일) 23시 59분까지 이다.
- 성적 제출 마감 제한으로 인해 Delay 기간이 없음에 유의.

Scoring

- 이 문제에서는 주어진 입력에 대해 여러 가지 출력이 가능하다. 즉, 이제까지의 과제와 다르게 정해진 답이 없다.
 - 아주 효율적인 답안도 가능하고 각 픽셀의 값을 하나씩 바꾸는 비효율적인 답도 가능하다.
- 따라서 출력한 답안의 효율성에 따라 점수가 부여된다.

Scoring

- 각 테스트 케이스에 대해

- 출력한대로 XOR 작업을 수행했을 때 재호가 원하는 그림이 나오지 않으면 0점.
- XOR 작업 개수(K)와 실제 출력한 XOR 작업의 개수가 다르면 0점.
- XOR 작업 개수(K)가 100,000보다 크면 0점.
- 총 XOR 작업의 사각형 넓이의 합이 20,000,000보다 크면 0점
- 올바르지 않은 XOR 작업을 출력하면 0점. (가령 범위를 벗어나거나 왼쪽 위, 오른쪽 아래 좌표가 아닐 경우)
- 실행 환경 제약(CPU 1초, 메모리 8MB) 을 어길 경우 0점.
- 올바른 출력을 했다면 $1 + 9 * \frac{\text{제출된 최소 XOR 작업의 개수}}{\text{출력한 XOR 작업의 개수}}$ 으로 점수가 주어진다.
- 즉 수강생 중 가장 좋은 답을 출력했다면 10점을, 가장 좋은 답변에 비해 아주 나쁜 답을 출력하면 1점을 받게 된다.

- 최종 점수는 각 테스트 케이스 점수들의 합으로 정해진다.

Test

- 테스트의 편의를 위해 putty를 통하여 프로그램을 채점할 수 있는 인터페이스를 구현해 두었으니 이를 잘 활용하도록 한다.
 - 이 명령어는 본 과목 실습 서버(prg1, prg2)에서만 사용 가능하다. 개인 노트북이나 martini 서버에서는 작동하지 않는다.
 - 이 인터페이스는 테스트의 편의를 위해 제공하는 것이므로 너무 의존하지 않도록 한다.
- 사용법: judge <실행 파일 이름>
 - ex) judge a.out
 - 소스 코드가 아니라 컴파일 하여 실행 가능한 파일의 위치를 입력으로 넣는다.
 - 이렇게 채점을 했더라도 moodle에 소스 코드를 제출해야 한다!

Test

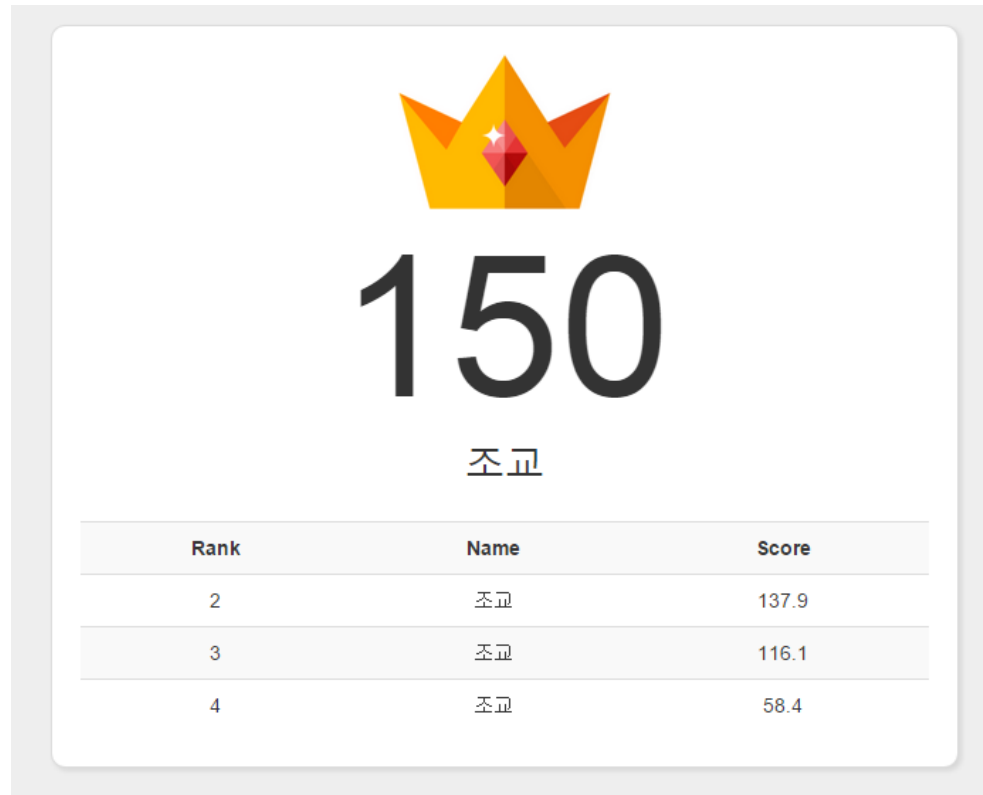
```
hcil@prg:~/SimpleJudge$ judge a.out
*****
You are hcil.

Testcase #01: OK, # of ops = 172, score = 4.9
Testcase #02: OK, # of ops = 209, score = 5.8
Testcase #03: OK, # of ops = 5223, score = 5.0
Testcase #04: OK, # of ops = 4995, score = 5.7
Testcase #05: OK, # of ops = 44967, score = 5.7
Testcase #06: OK, # of ops = 45047, score = 5.7
Testcase #07: OK, # of ops = 44872, score = 5.8
Testcase #08: OK, # of ops = 125454, score = 0.0
Testcase #09: OK, # of ops = 124718, score = 0.0
Testcase #10: OK, # of ops = 125227, score = 0.0
Testcase #11: OK, # of ops = 125, score = 6.9
Testcase #12: OK, # of ops = 788, score = 4.3
Testcase #13: OK, # of ops = 4932, score = 4.9
Testcase #14: OK, # of ops = 8834, score = 3.7
Testcase #15: OK, # of ops = 123501, score = 0.0

Your score is 58.
Your rank is 4/4.
```

Test

- 또한 현재 상위 6명의 점수를 <http://147.46.241.158:8081/> 에서 확인할 수 있다.
 - 주소는 moodle 서버와 같고 포트번호만 + 1



Tips

- 이 문제는 작업의 개수가 최소가 되지 않더라도 부분 점수를 받을 수 있다.
- 따라서, 비효율적이더라도 주어진 그림을 만들 수 있는 알고리즘을 만들고 이를 개선해가자.
- 몇몇 케이스에서만 Wrong Answer가 나오는 경우
 - 총 XOR 작업의 사각형 넓이의 합이 20,000,000보다 적어야 한다는 조건을 위반하는 경우일 확률이 높다.
 - $XOR(1, 4, 3, 5)$ 는 넓이가 $(3 - 1 + 1) * (5 - 4 + 1) = 3 * 2 = 6$ 인 사각형의 색을 바꾼다.