

# Object-Oriented Programming Report

Assignment 1-2

Professor	Donggyu Sim
Department	Computer engineering
Student ID	2022202093
Name	Wonju moon
Class (Design / Laboratory)	1 / B (미수강시 0로 표기)
Submission Date	2023. 3. 24

서식 지정함: 글꼴: (영어) 한컴 바탕, (한글) 한컴 바탕,  
글꼴 색: 텍스트 1

서식 있음: 간격 없음, 줄 간격: 1줄, 눈금에 맞춤,  
텍스트 맞춤: 자동

## Program 1

### □ 문제 설명

숫자를 두 개 입력 받아서 각 숫자 한 자리 수 단위로 나눠준다. 그 후 처음 글자부터 출력을 한 후 전체 글자 배열에서 해당 글자가 중복되는 만큼 숫자를 세서 해당 숫자 바로 뒤에 출력한다. 이때, 마지막 배열까지 비교할 때 해당 숫자가 이전에 이미 출력됐다면 중복출력을 하지 않고 넘어간다.

### □ 결과 화면

```
C# 선택 Microsoft Visual Studio 디버그 콘솔
input num1 :
79
input num2 :
897687543217
749282615141312111
C:\Users\82104\OneDrive\바탕 화면\2021\2학기\컴퓨터구조\Lab\Lab1\Program1\Program1.cs(1,1): The type or namespace name 'Program1' does not exist in the namespace 'Lab1' (are you missing a using directive or an assembly reference?)
```

```
C# Microsoft Visual Studio 디버그 콘솔
input num1 :
897687543217
input num2 :
897687543217
849276625242322212
(C:\Users\82104\OneDrive\바탕 화면\객프실\1학기\Lab1\Program1\Program1.cs(1,1): The type or namespace name 'Program1' does not exist in the namespace 'Lab1' (are you missing a using directive or an assembly reference?))
이 창을 닫으려면 아무 키나 누르세요...
```

두 수를 char 배열로 입력 받은 후 아스키코드를 이용하여 int 형으로 바꿔준다. 그리고 int num 배열에 있는 숫자를 앞에서 인식하며 0으로 초기화 시킨 int bucket 함수에 해당하는 인덱스 값을 1 씩 증가시킨다. 예를 들어 int num[0] == 7 이라면 bucket[7] += 1 을 적용한다. 이러한 원리로 두 배열에 있는 모든 숫자의 개수를 세준 후 num 숫자와 bucket 을 돌아가면서 출력한다. 이때 반복문을 사용해서 num[k] 인덱스 값과 num[0]~num[k-1]까지의 값 중 하나라도 같다면 출력하지 않는다.

## □ 고찰

숫자를 바로 int 형 배열로 입력받으면 인덱스 값 하나에 모든 수가 저장되기 때문에 char 형으로 받은 후 아스키코드를 이용해서 한 글자씩 끊어서 다시 int 형 배열에 입력해준다는 생각이 어려웠음. 또한, 중복을 비교할 때 cnt 라는 새로운 함수를 선언해서 cnt 를 하나씩 쌓아준 후 해당 값을 return 해줬다. 전달받은 cnt 값이 양수라면 하나라도 중복이 됐다는 뜻이므로 출력하지 않았음.

처음 int num 배열을 0 으로 초기화하면 bucket[0]의 값이 굉장히 많이 증가했음. 이러한 오류를 잡기위해 num 을 초기화하지 않았고, 더 확실하게 오류를 없애기 위해 num 을 사용하는 모든 조건문에 num 의 인덱스가 [0,9]사이의 값을 가져야 한다는 조건을 걸었음.



## Program 2

### □ 문제 설명

Sender함수와 Transmission channel 함수와 receiver함수를 만들어 CRC를 구현하는 문제임.

CRC란 순환 중복 검사로 데이터를 전송할 때 데이터에 오류가 있는지 확인하기 위한 오류 감사 기능임. 데이터를 전송하기 전에 전달할 데이터 값에 CRC값을 덧붙여 전송하고, 전송받은 데이터 값으로 다시 CRC를 계산하여 CRC값이 0이 아닌 경우 데이터 전송 과정 중에 오류가 발생했다는 것을 확인할 수 있음. Sender함수에서 사용자로부터 12비트의 데이터를 입력받아 해당 데이터를 XOR연산하고 나머지를 구하여 해당 나머지를 receiver로 전달함. 여기서 CRC값을 다시 확인하여 0인 경우 에러가 발생하지 않았다고 판단함. 그러나 모든 에러를 잡을 수는 없음

### □ 결과 화면

죄송합니다. 구현하지 못했습니다.

### □ 고찰

죄송합니다. 구현하지 못했습니다.



## Program 3

### □ 문제 설명

프로그램을 실행하면 열고싶은 파일의 이름을 적고 해당 파일을 파일입출력을 통해 연다. 그 후 패턴인식을 몇 개를 받을 지 입력받는다. 해당 숫자만큼 패턴을 입력받고 해당 패턴이 txt 파일 내에 적혀있다면 순서대로 출력한다. 이때, \*을 입력하면 \*을 제외하고 입력된 패턴이 txt 파일에 있다면 모두 출력하고, ?를 입력하면 txt 파일 내에 해당 ?자리를 제외한 다른 자리가 모두 같고 입력한 글자수와 비교할 글자수가 같다면 순서대로 출력한다.

## □ 결과 화면

파일(F) 편집(E) 시각(V) 도구(T)

```
hello_world.cpp
oop_assignment1.zip
oop_assignment2.zip
hello.txt
oop_assignment3.zip
```

C:\Users\82104\OneDrive\바탕 화면\객포실\1주차\1-2

```
input file name : filename_list.txt
no
```

Microsoft Visual Studio 디버그 콘솔


```
input file name : filename_list.txt
2
hello*
oop_assignment?.zip

hello_world.cpp
hello.txt
oop_assignment1.zip
oop_assignment2.zip
oop_assignment3.zip

C:\Users\82104\OneDrive\바탕 화면\객포실\1주차\1-2
개).
이 창을 닫으려면 아무 키나 누르세요...
```

파일이름을 입력받아 파일을 열고 반복횟수를 정한 후 compare 함수에 대입함. 최종적으로 파일명을 출력할 2 차원 배열인 finalname 을 선언함. getline 함수를 통해서 char 형 data 배열에 txt 파일에 적힌 글을 한 줄씩 입력받고, 사용자로부터 입력받은 패턴은 char 형 name 배열에 입력함.

strcmp 함수를 통해서 name 배열과 data 배열을 비교하며 name 에 있는 내용이 data 에 있다면 data 배열을 finalname 에 복사하고 마지막에 finalname 을 한번에 출력함.



이때, name 배열에 \*이 있다면 그 앞까지 글자수를 카운트해서 strncmp 를 통해 해당 글자수만큼만 배열을 비교한다. 만약 같다면 \* 앞까지의 글자가 data 에 있는 문장과 일치한다고 판단하고 finalname 에 입력함.

그리고 중간에 ?가 입력된다면 ?를 기준으로 ?를 제외한 앞 글자와 뒷 글자들을 모두 비교해주고 함수의 길이가 같고 비교하는 모든 글자 수가 같다면 finalname 에 입력함.

그 후 최종적으로 finalname 을 출력함.

## □ 고찰

다른 부분보다 \*와 ?를 어떻게 처리할지가 문제였음.

첫 번째 문제) strcmp 로는 ?뒤 배열을 비교할 수 없었음.

이러한 경우 결국 반복문을 통해서 인덱스 값 하나하나 비교하는 방법으로 해결함.

두 번째 문제) 인덱스 값을 하나하나 비교하다 보니 문자 하나가 같다고 문자열 전체가 같다는 판단을 해버리는 경우가 생김.

name 에 입력된 문자 중에 ?를 제외한 문자열의 길이를 세고 data 에 입력된 문자열과 비교함. ?자리의 값을 제외한 모든 인덱스 값을 비교하면서 같은 경우 q 를 1 씩 증가함. 그래서 1) 배열의 길이가 같고 2) q 와 문자열의 길이가 같은 경우를 finalname 에 입력. 이러한 방식으로 ?에 관한 문제를 해결해줬음..

세 번째 문제) \*을 구현하는 과정에서 strcmp 를 이용하여 글자수 비교에 어려움을 느낌. 이때 \*앞까지의 글자수를 따로 센 후 strcmp 를 이용하여 \* 앞자리까지만 비교를 해서 \*앞까지 모든 글자수가 같다고 판단하는 방식으로 해결함.

네 번째 문제) \*이 문자 중간 혹은 맨 앞에 있는 경우 \* 뒤를 비교할 때 어려움을 느낌. ?와 달리 \*은 문자의 길이가 달라도 적용되어야 하므로 ?처럼 인덱스 하나하나를 비교하기도 어렵고 그렇다고 strcmp 를 쓰기에는 \* 뒤에 있는 문자열을 비교할 수 없기에 사용을 못함. 해당 문제는 해결하지 못함. 이러한 경우 hello\*을 입력하면 패턴을 잘 찾지만 \*hello 를 입력하면 패턴을 찾지 못함.



## Program 4

### □ 문제 설명

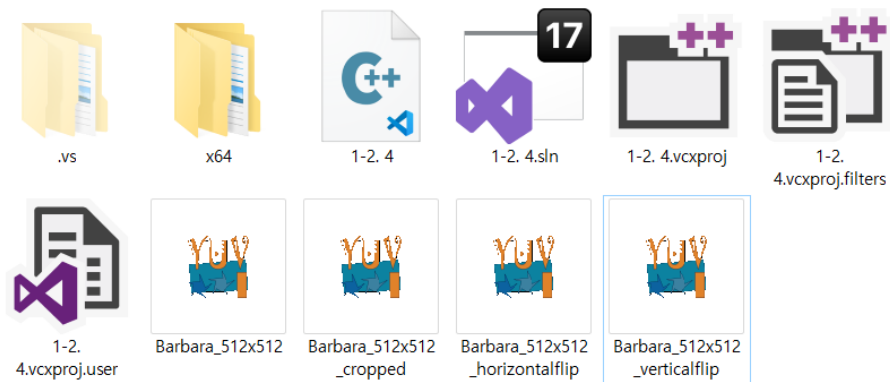
512x512크기의 원본 이미지파일을 가지고 원본 파일을 자른 이미지 파일 하나와 상하반전 시킨 이미지 파일 하나 그리고 좌우반전 시킨 이미지 파일 하나까지 총 3개의 이미지파일을 만든다. 이때 원본 파일을 잘라 만든 이미지 파일의 파일명에는 잘라서 만든 이미지 파일의 높이와 너비가 포함된다. 또한 상하반전 시킨 이미지 파일의 파일명에는 원본파일의 이름과 verticalflip이라는 문장이 함께 들어가고 좌우반전 시킨 이미지 파일의 파일명에는 원본파일의 이름과 horizontalflip이라는 문장이 함께 들어간다.




## □ 결과 화면

```
Microsoft Visual Studio 디버그 콘솔
Input file name: barbara_512x512.raw
1st coordinate : 256, 0
2nd coordinate : 512, 192

C:\Users\82104\OneDrive\바탕 화면\객프실\1주차\1-2
개).
이 창을 닫으려면 아무 키나 누르세요...
```



동작할당으로 input 과 output 배열을 만들어준 후 파일 입출력을 통해서 input 배열에 원본 사진을 가져온다. 그 후 input 배열을 output 복사한 후 output 배열을 가공하여 출력.



첫 번째 경우) 사진을 자르는 경우

사진을 좌표처럼 인식하여 왼쪽 위끝 모서리를 (0,0) 오른쪽 아래끝 모서리를 (511,511)로 둬. 그 후 좌표를 두개 입력받고 더 큰 좌표값에서 작은 좌표값을 빼는 방법을 통해서 범위를 정해줌. 반전의 경우에는 반복문을 통해서 상하반전의 경우 중심 가로축을 기준으로 좌우 반전의 경우에는 중심 세로축을 기준으로 대칭시켜 출력함.

## □ 고찰

파일의 크기는 정사각형인데 좌우반전과 상하반전의 2차원 배열의 조건이 조금 달랐음. 좌우반전과 달리 상하반전의 경우 열에서 -1을 해줘야했음. 둘 다 메커니즘은 같은데 이유를 알 수 없었음. 아마 배열의 끝에 있는 null 값 때문이라는 결론을 도출함.

파일을 자를 때 높이와 너비를 입력받고 높이와 너비를 파일 이름에 입력해야 하는데 입력 방법에서 어려움을 느껴 구현하지 못함.

입력값이 (256,0) 그리고 (512,192)였는데 512x512의 크기에서 인덱스는 행과 열의 끝이 511인데 어떻게 512를 입력받을 수 있는지 알 수가 없었음. 오류가 나지 않아서 넘어감.