LAB2

Lab2

- The deadline for lab2 submission is 15th March at 11:59 pm.

- Folder name : lab2
- Code name : p2-1.c, p2-2.c
- Each code will be tested by 5 different inputs (files).

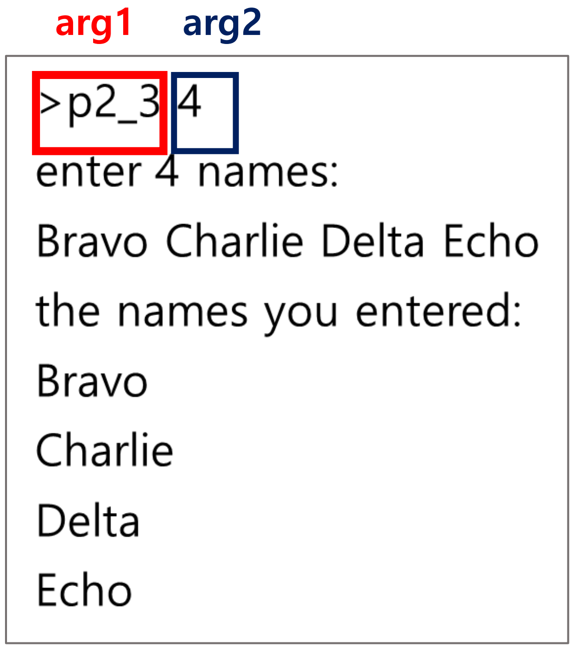
Evaluation criteria

Category	Evaluation	
2-1	50	
2-2	50	
Total	100	

- *Use GCC **9.3** version.*
- *No score will be given if the gcc version is different.*

Lab 2-1. Dynamic allocation

Use a command line argument for the total number of students ($n>2$). In the standard input, get n(the number you input) names in a single line when the instruction “enter names:” is given. Use 2D dynamic allocation to store all names in one variable. Print your result in standard output.



```
>p2_3 4
enter 4 names:
Bravo Charlie Delta Echo
the names you entered:
Bravo
Charlie
Delta
Echo
```

Lab 2-1. Dynamic allocation

Command line sample

```
$ filename arg1 arg2 arg3
```

int main(int argc, char *argv[])

Filename	arg1	arg2	arg3
argv[0]	argv[1]	argv[2]	argv[3]

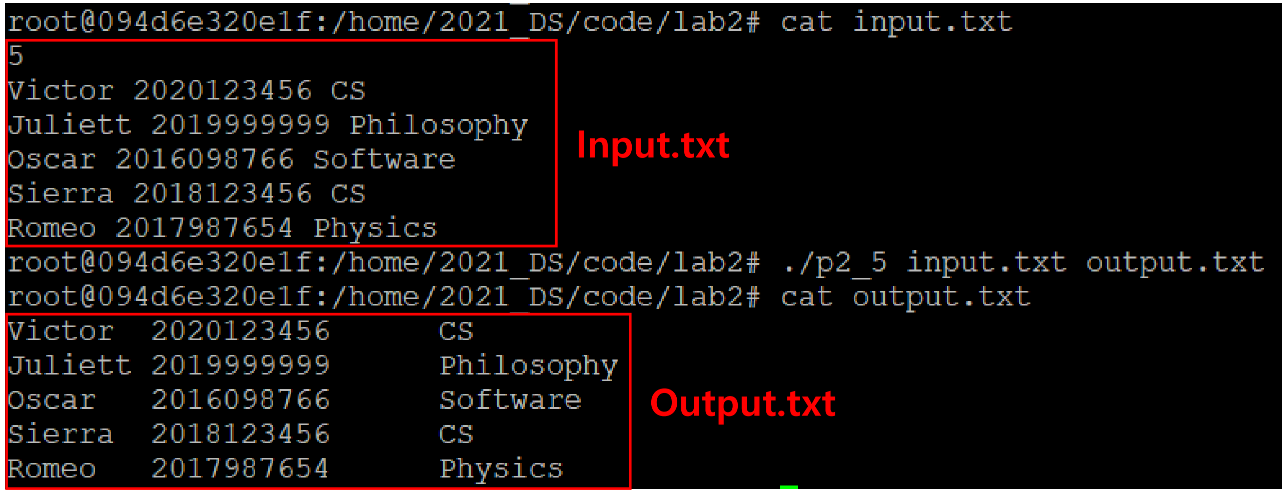
argc=4

Lab 2-1. Dynamic allocation

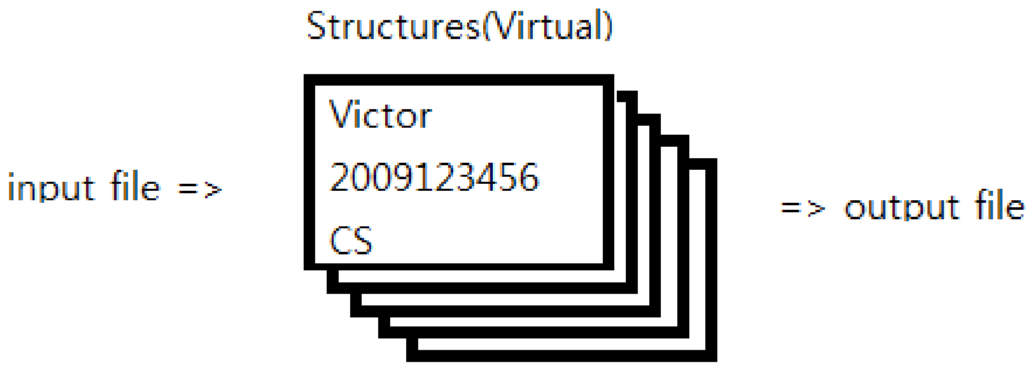
- program name : p2_1.c
- data structure : array of pointers
- input : n names
- output : n names
- condition:
  - *the number of students should be more than two.*
  - *use dynamic allocation to store names (malloc())*
  - *the length of the name should be up to 30 characters.*
  - *no blank space is allowed in the name.*

Lab 2-2. Structure

Use a command line argument with a designated input file and your own output file. Read the first integer from the input file to figure out the number of students. Create an array of structures and put the information in it. Then print the structures in the output file.



```
root@094d6e320e1f:/home/2021_DS/code/lab2# cat input.txt
5
Victor 2020123456 CS
Juliett 2019999999 Philosophy
Oscar 2016098766 Software
Sierra 2018123456 CS
Romeo 2017987654 Physics
root@094d6e320e1f:/home/2021_DS/code/lab2# ./p2_5 input.txt output.txt
root@094d6e320e1f:/home/2021_DS/code/lab2# cat output.txt
Victor  2020123456      CS
Juliett 2019999999      Philosophy
Oscar   2016098766      Software
Sierra  2018123456      CS
Romeo   2017987654      Physics
```



Lab 2-2. Structure

- program name: p2_1.c
- data structure : structure

```
typedef struct student{
        char *name;
        int student_id;
        char *major;
}student;
```

- input : the number of students and their information
- output : all of the student information
- conditions :
  - the length of name and major should be up to 30 characters
  - no blank space is allowed in the name or major

Lab 2-2. Structure – sample code

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>

typedef struct student{
	char *name;
	int student_id;
	char *major;
}student;
```

```
int main(int argc, char** args)
{
	char *input_path, *output_path;
	FILE *fi, *fo;

	int num;
	int i;

	student* students;

	input_path = args[1];
	output_path = args[2];

	fi = fopen(input_path, "r");
	fo = fopen(output_path, "w");

	....
}
```