

자료구조

실습 진행 및 Lab01 추가설명

KHU NLP Lab.

앞으로의 Lab 진행

- <http://www.jblearning.com/> 에서 받은 코드로 진행 안함
 - 버전 문제로 코드 내용이 다름
- e-campus 업로드된 참고자료 코드로 한 학기 동안 실습 진행
- 제출 형식 : 학번_이름_LAB##.zip
 - ex) 2020111111_홍길동_LAB01.zip

Lab01-01 Case Study

1. e-campus에 업로드된 파일을 받아서 압축을 풀기
2. Chapter02 폴더에 있는 Application과 String test 폴더 중 하나를 선택해서 안에 있는 모든 파일들을 복사해서 컴파일러 프로젝트 폴더 안에 위치 시킴
3. 인코딩 진행
4. 컴파일

Lab01 - 02

- Lab01-7page

```
typedef char String[9];  
struct StudentRecord  
{  
    String firstName;  
    String lastName;  
    int id;  
    float gpa;  
    // char gender;  
    int currentHours;  
    int totalHours;  
};  
StudentRecord student;  
StudentRecord students[100];
```

선언된 StudentRecord 구조체

	Length	Offset
firstName	9	0
lastName	9	9
Id	4	18
gpa	4	22
currentHours	4	26
totalHours	4	30

[Offset Table]

[offset Table 구조체 크기]

34

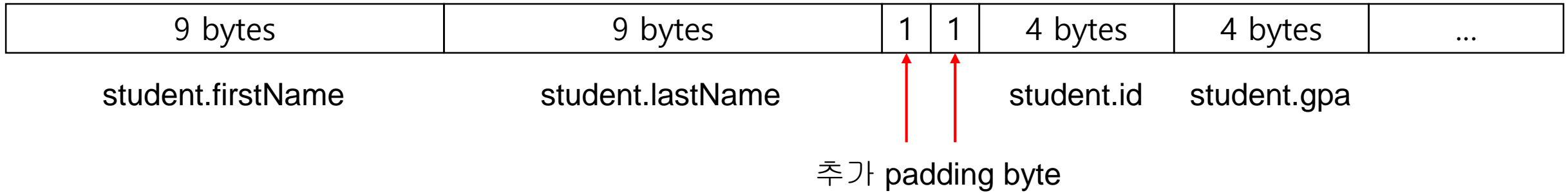
[Visual Studio 구조체 크기]

36

구조체 크기 차이가 발생

Visual studio 컴파일러 출력 결과

```
Microsoft Visual Studio 디버그 콘솔
Base Address : 5242372 size of student : 36
fristName Address : 5242372 size of firstName : 9
lastName Address : 5242381 size of lastName : 9
id Address : 5242392 size of id : 4
gpa Address : 5242396 size of gpa : 4
currentHours Address : 5242400 size of currentHours : 4
totalHours Address : 5242404 size of totalHours : 4
```



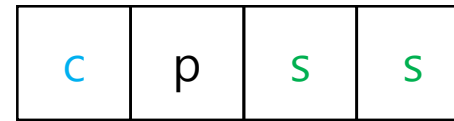
Struct byte Padding

- Struct Padding
 - 구조체 멤버들을 메모리에 할당 하는 규칙
 - 컴파일러에 Struct Padding 규칙이 정해져 있음
 - 기본적으로 크기가 가장 큰 멤버 변수를 기준으로 모든 멤버 변수의 메모리 크기를 맞춤
 - OS 마다 규칙이 다르게 적용
 - ex) 32bit or 64bit

Struct byte Padding

- 기본적으로 크기가 가장 큰 멤버 변수를 기준으로 모든 멤버 변수의 메모리 크기를 맞춤

```
struct Test
{
    char c;
    short s;
}
```



추가 padding byte

Struct byte Padding

- Char array??
 - char 과 char[] 특별히 이전 멤버 메모리 뒤에 패딩을 하지 않음
- 가장 큰 멤버 변수 크기가 최대 default byte 크기 큰 경우
 - ex) 최대 default byte : 8
 - 4의 배수로 단위로 bytes padding을 진행

```
typedef char String[9]
struct Test
{
    String a;
    String b;
    long long s;
}
```

$\text{sizeof}(a) + \text{sizeof}(b) + \text{padding 2 bytes} + \text{sizeof}(s) = 9 + 9 + 2 + 8$