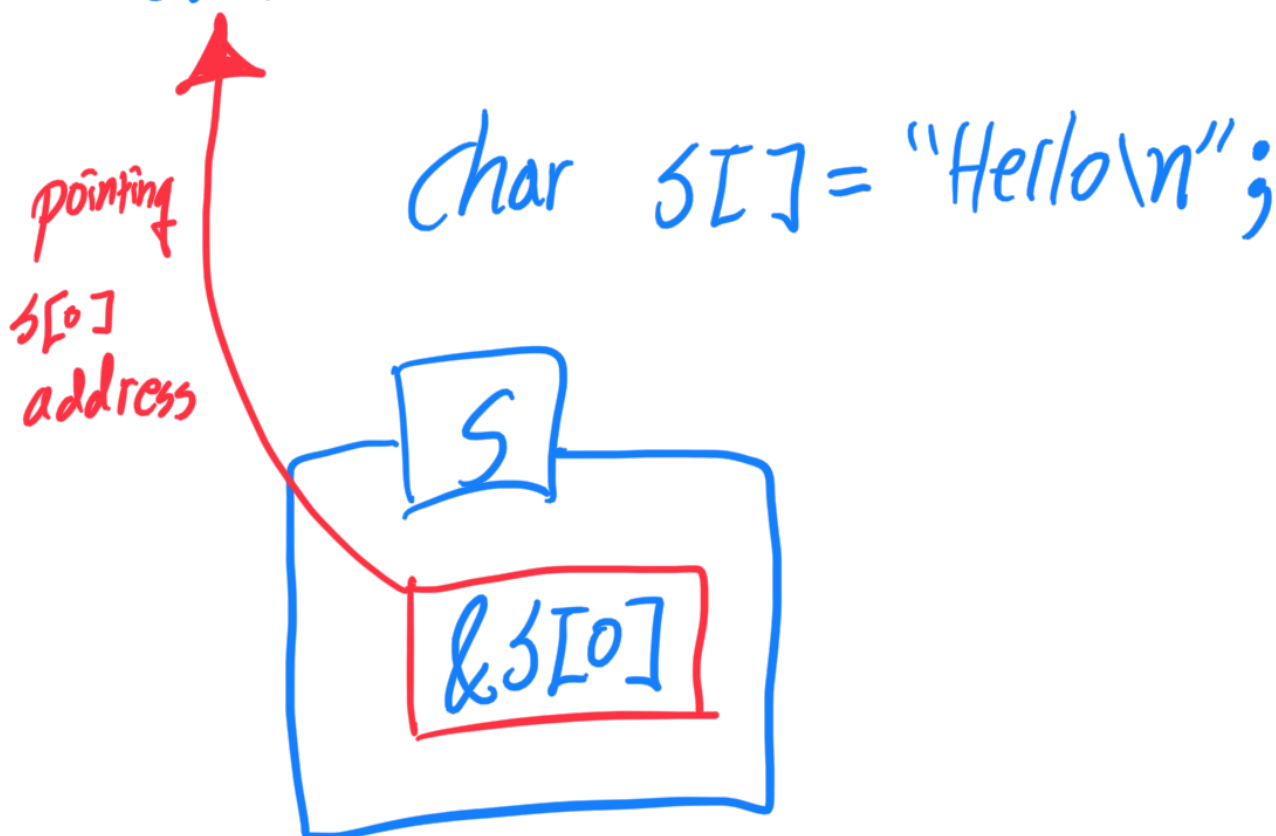
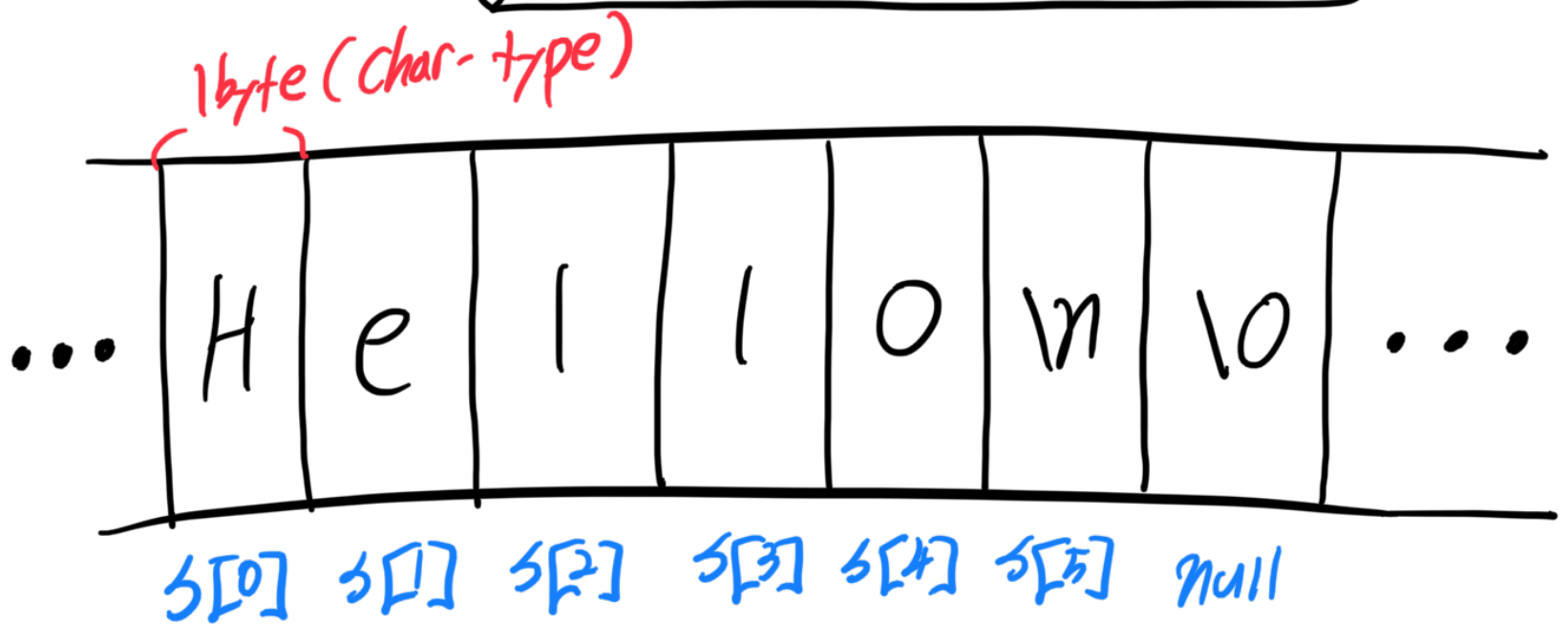


C language

pointer with Argument

Using char s[]



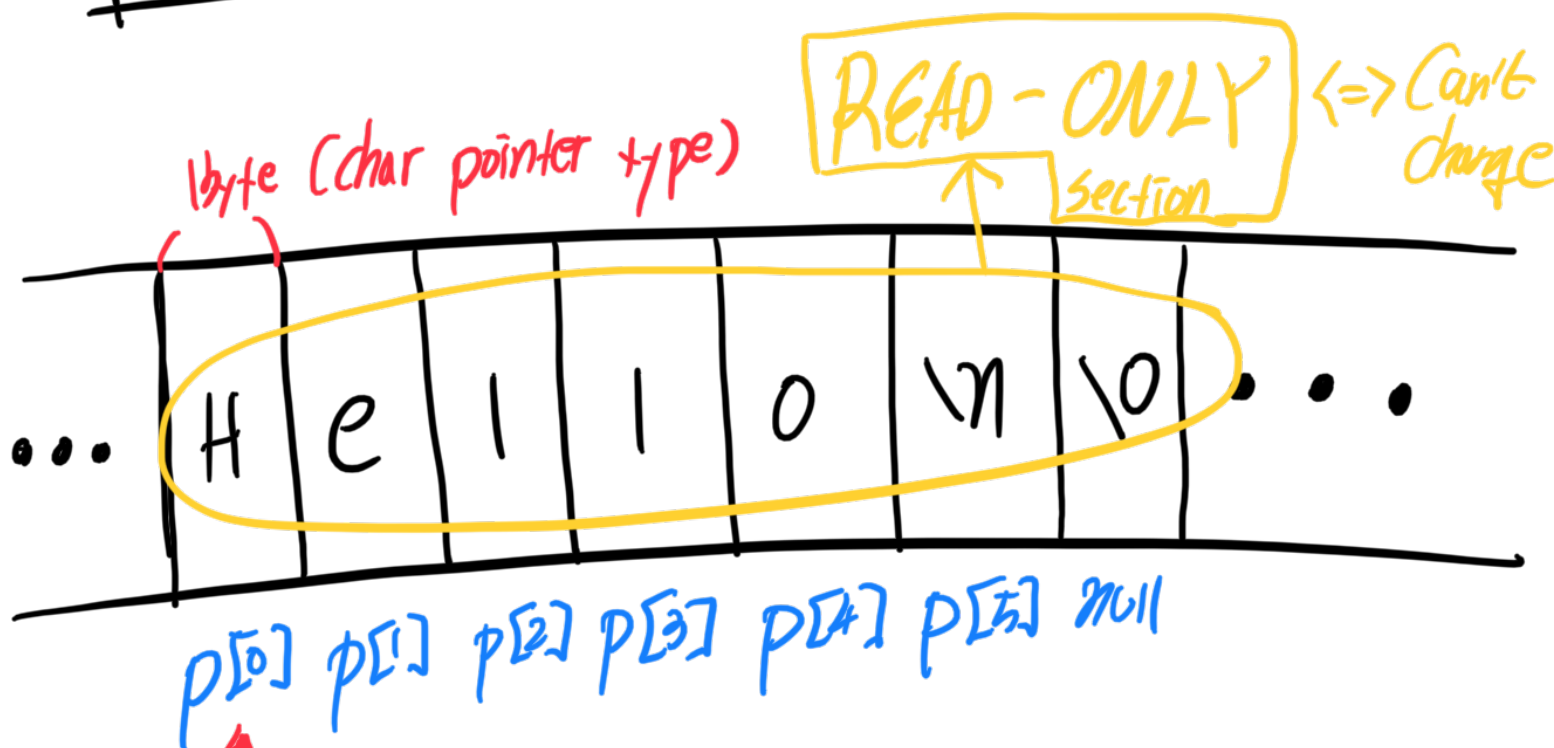
s = &s = &s[0] = initial address

`printf("%s", s);`

↳ %s wants initial address of array s.

<Remark>

Using `char* p`



`char* p = "Hello\n";`

pointing
`p[0]` address




Char s[] = "Hello\n";

= {'H', 'e', 'l', 'l', 'o', '\n'}



Each components are
saved in memory location, which
we can read and write

Char* p = "Hello\n";



Each components are
saved in memory location,
which we can only read.

< Difference >

Pointer in parameter and argument

Void print1(char str1[]) {

}

and

Void print2(char* str2) {

}

↑
parameter



are interpreted exactly same.

char str1[] = char* str1

char* str2 = char str2[]

(just, as parameter)

```

int main() {
    ...
    print1(p);
    print1(5);
    ...
}

```

argument

```

Void print1(char s1[]){
    s1[i]='a';
    ↳ not allowed!
    Because argument
    is char * p
    Read-only
}

```

```

Void print1(char s1[]){
    s1[i]='a';
    ↳ allowed!
    Because argument
    is char s1[]
    Read & Write
}

```