C Programming (W12)

Welcome!!

Please check attendance individually.

(Mobile App)

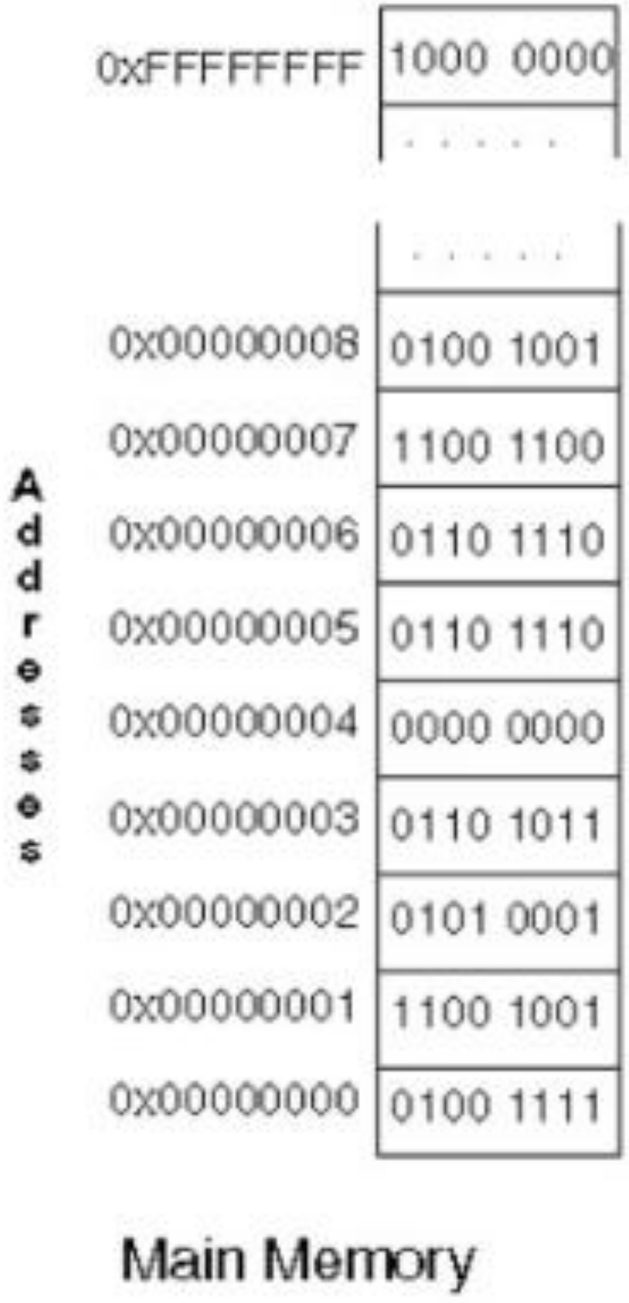
miricanvas.com

Things to do today

01 Lecture Notes (Ch.13)
- Structure (. & -> operator)

02

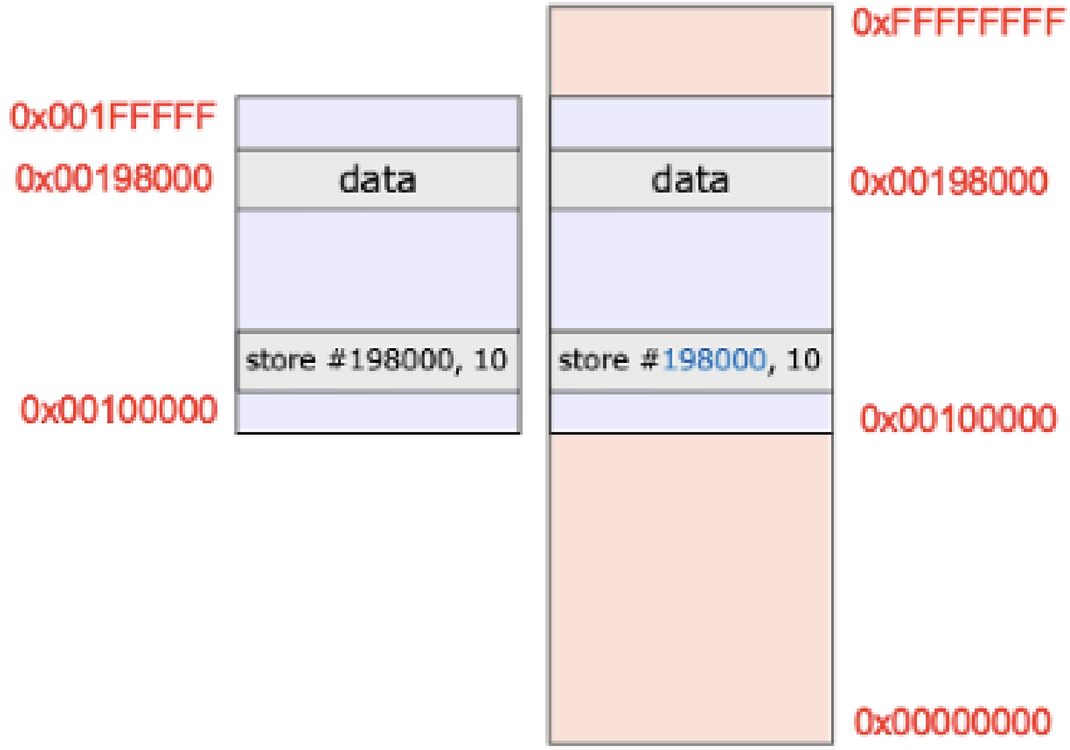
03



- Compile time

```
void func( )
{
  int data;

  ...
  data = 10;
  ...
}
```



A pointer represents an address

char s[] = "abc";

Index	Value	Expression	Address Expression
0	`'a'`	`s[0]`	`&s[0]`
1	`'b'`	`s[1]`	`&s[1]`
2	`'c'`	`s[2]`	`&s[2]`
3	`'\0'`	`s[3]`	`&s[3]`

- **Values:**
  `s[0]` → `'a'` `s[1]` → `'b'` `s[2]` → `'c'` `s[3]` → `'\0'` (null terminator)

- **Addresses:**
  `&s[0]`, `&s[1]`, `&s[2]`, etc. — sequential in memory

- **Pointers:**

```
char *p = &s[0];  // or simply: char *p = s;
```

  - Now: `p == s`
  - You can access the characters via pointer arithmetic:
    - `*p` → `'a'`
    - `*(p + 1)` → `'b'`
    - `*(p + 2)` → `'c'`

Array & Pointer

int a = 10;

value : a

address : &a

pointer : int *p = &a

p is address AND *p is value at this address

int a[3] = { 10, 20, 30 };

value : a[0] / a[1] / a[2]

address : &a[0] / &a[1] / &a[2]

pointer : int *p0 == &a[0]

int *p1 == &a[1]

int *p2 == &a[2]

*(p + 0) == 10

*(p + 1) == 20

*(p + 2) == 30

char s[] = “abc”;

value : s[0] / s[1] / s[2]

address : &s[0] / &s[1] / &s[2]

pointer : char* p0 == &s[0]

See you next week!
DO NOT miss the classes

miricanvas.com