시스템 프로그래밍을 위한 C언어

Function 포인터의 활용 및 필요성

현대자동차 입문교육
박대진 교수

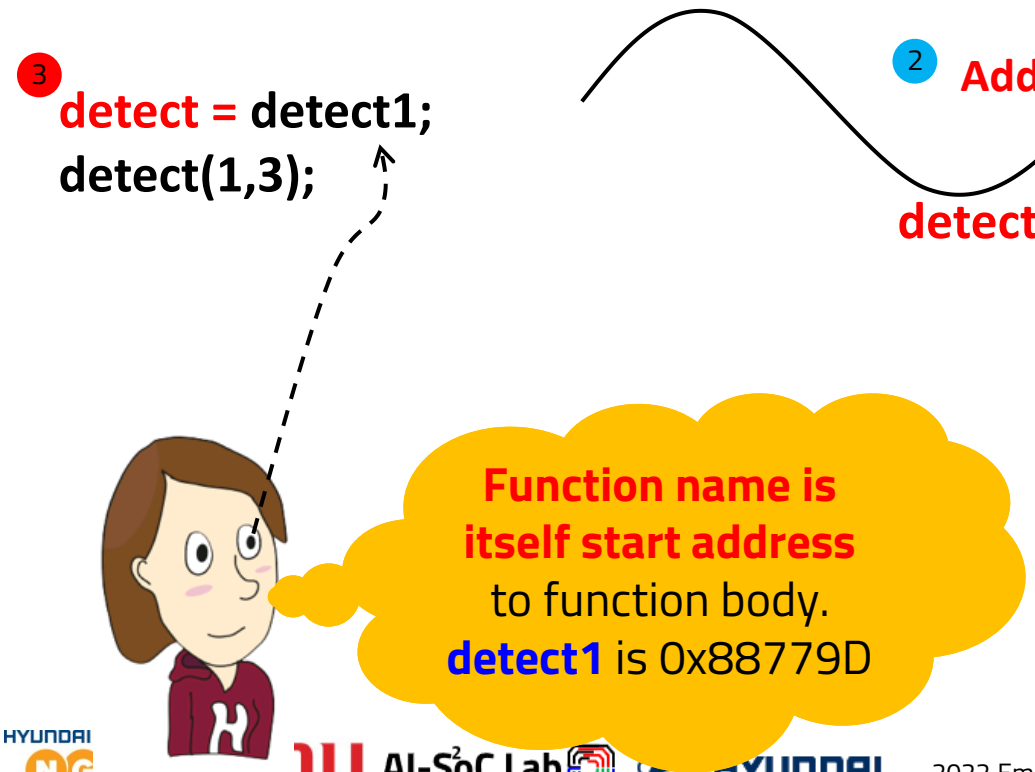
Pointer to Function

<Function to Body>

```
int detect1(int a, char b) {
   // do something
}
```

<Pointer to Function>

```
int (*detect)(int, char);
```



Function is also allocated in memory

Replacing Function Start Address

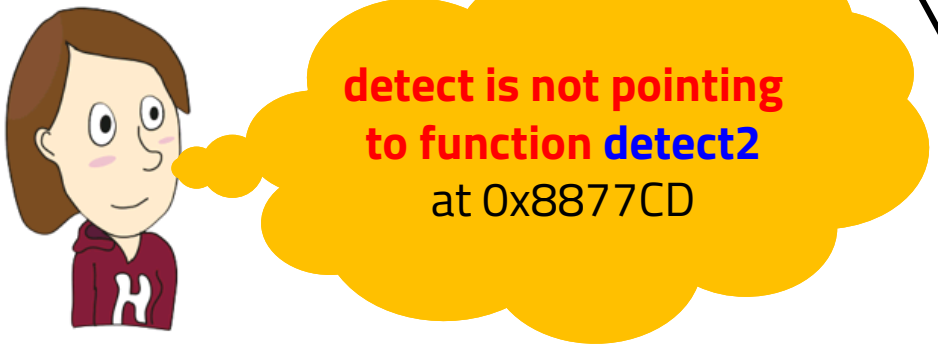
<Function to Body>

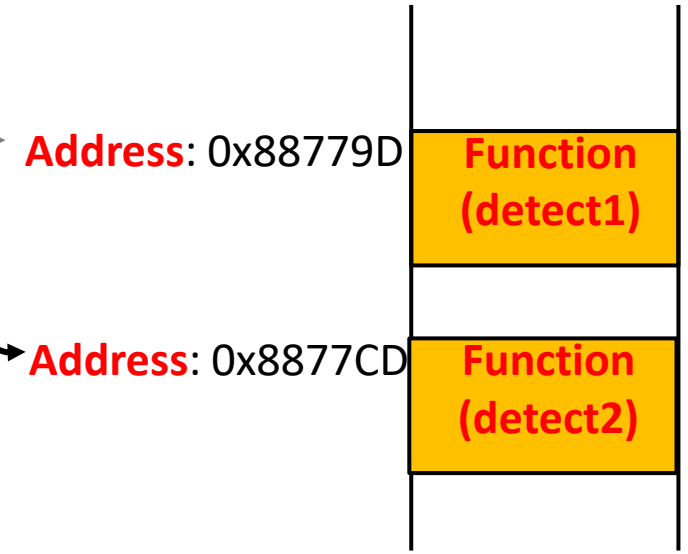
```
int detect1(int a, char b) {
    // do something
}
int detect2(int a, char b) {
    // do something
}
```

<Pointer to Function>

```
int (*detect)(int, char);

detect = detect1;
detect(1,2);
detect = detect2;
detect(3,4);
```





Reconfiguring Functions using Function Pointer Redirection

```c
void detect_v1(int a, char b) {
    // do something
    printf("detect_v1() is activated\n");
}
void detect_v2(int a, char b) {
    // do something
    printf("detect_v2() is activated\n");
}
void (*detect)(int, char) ;
```

```c
void main() {
    int k=10;
    char c=12;
    int cond=1;
    if(cond==1)
        detect = detect_v2;
    else
        detect = detect_v1;
    detect(10, 12);
}
```