

```
//  
// Solution to C function pointers primer by  
// Golden G. Richard III (@nolaforensix).  
//
```

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

```
typedef struct funcs {  
    int (*openit)(char *name, int prot);  
    void (*closeit)(void);  
} funcs;
```

```
int myopenit(char *name, int prot) {  
    printf("myopenit() called.\n");  
    return 0;  
}
```

```
void mycloseit(void) {  
    printf("mycloseit() called.\n");  
}
```

```
static funcs f=  
{  
    .openit=myopenit,  
    .closeit=mycloseit,  
};
```

```
void init_funcs(funcs *f) {  
    (*f).openit=myopenit; // or  
    f->openit=myopenit;  
    (*f).closeit=mycloseit; // or  
    f->closeit=mycloseit;  
}
```

```
int main(int argc, char *argv[]) {  
    funcs g;  
    funcs *h;  
    funcs another;  
  
    init_funcs(&another);  
  
    h=malloc(sizeof(funcs));
```

# DO NOT DISTRIBUTE

```
h->openit=myopenit;  
h->closeit=mycloseit;  
  
g.openit=myopenit;  
g.closeit=mycloseit;  
  
f.openit(NULL, 0);  
}
```