

# C Programming

## Day18.B

2017.11.14

4 structure definitions

Copyright (c) 2015 - 2017 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

```

#include <stdio.h>

struct aaa {
    int a;
    int b;
};

struct bbb {
    double a;
    double b;
};

void pr_aaa( struct aaa X, char *s ) {
    printf("-----\n");
    printf("%s.a = %d \n", s, X.a);
    printf("%s.b = %d \n", s, X.b);
}

void pr_bbb( struct bbb X, char *s ) {
    printf("-----\n");
    printf("%s.a = %g \n", s, X.a);
    printf("%s.b = %g \n", s, X.b);
}

int main(void) {

    struct aaa A = { 100, 200 };
    struct bbb B = { 11.1, 22.2 };
    struct bbb C;

    C = B;

    pr_aaa( A, "A");
    pr_bbb( B, "B");
    pr_bbb( C, "C");
}

```

```

-----
A.a = 100
A.b = 200
-----
B.a = 11.1
B.b = 22.2
-----
C.a = 11.1
C.b = 22.2

```

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

```
struct bbb {  
    double a;  
    double b;  
};
```

Structure Returning function

```
struct bbb ADD_bbb( struct bbb X, struct bbb Y) {  
    struct bbb S;  
  
    S.a = X.a + Y.a;  
    S.b = X.b + Y.b;  
    return(S);  
}
```

```
int main(void) {  
    struct bbb B = { 11.1, 22.2 };  
    struct bbb C;  
  
    C = B;  
  
    printf("\nADD_bbb( B, C )=====\n\n");  
    printf("B.a= %f \n", B.a);  
    printf("B.b= %f \n\n", B.b);  
  
    printf("C.a= %f \n", C.a);  
    printf("C.b= %f \n\n", C.b);  
  
    C = ADD_bbb( B, C );  
  
    printf("C.a= %f \n", C.a);  
    printf("C.b= %f \n", C.b);  
}
```

```
ADD_bbb( B, C )=====
B.a= 11.100000
B.b= 22.200000

C.a= 11.100000
C.b= 22.200000

C.a= 22.200000
C.b= 44.400000
```

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

```
struct bbb {  
    double a;  
    double b;  
};
```

```
typedef struct bbb BT;
```

```
BT ADD_bbb( BT X, BT Y) {  
    BT S;  
  
    S.a = X.a + Y.a;  
    S.b = X.b + Y.b;  
    return(S);  
}
```

```
int main(void) {  
    BT B = { 11.1, 22.2 };  
    BT C;  
  
    C = B;  
  
    printf("\nADD_bbb( B, C )=====\\n\\n");  
    printf("B.a= %f \\n", B.a);  
    printf("B.b= %f \\n\\n", B.b);  
  
    printf("C.a= %f \\n", C.a);  
    printf("C.b= %f \\n\\n", C.b);  
  
    C = ADD_bbb( B, C );  
  
    printf("C.a= %f \\n", C.a);  
    printf("C.b= %f \\n", C.b);  
}
```

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

```
struct bbb {  
    double a;  
    double b;  
} B = {11.1, 22.2}, C;
```

*global variables*

```
typedef struct bbb BT;
```

```
BT ADD_bbb( BT X, BT Y) {  
    BT S;  
  
    S.a = X.a + Y.a;  
    S.b = X.b + Y.b;  
    return(S);  
}
```

```
int main(void) {  
  
    C = B;  
  
    printf("\nADD_bbb( B, C )=====\n\n");  
    printf("B.a= %f \n", B.a);  
    printf("B.b= %f \n\n", B.b);  
  
    printf("C.a= %f \n", C.a);  
    printf("C.b= %f \n\n", C.b);  
  
    C = ADD_bbb( B, C );  
  
    printf("C.a= %f \n", C.a);  
    printf("C.b= %f \n", C.b);  
  
}
```

```
typedef struct bbb { double a; double b; }  
BT;
```

```
#include <stdio.h>  
  
typedef struct bbb {  
    double a;  
    double b;  
} BT;  
  
BT ADD_bbb( BT X, BT Y) {  
    BT S;  
  
    S.a = X.a + Y.a;  
    S.b = X.b + Y.b;  
    return(S);  
}  
  
int main(void) {  
    BT B = {11.1,22.2}, C;  
  
    C = B;  
  
    printf("\nADD_bbb( B, C )=====\n\n");  
    printf("B.a= %f \n", B.a);  
    printf("B.b= %f \n\n", B.b);  
  
    printf("C.a= %f \n", C.a);  
    printf("C.b= %f \n\n", C.b);  
  
    C = ADD_bbb( B, C );  
  
    printf("C.a= %f \n", C.a);  
    printf("C.b= %f \n", C.b);  
  
}
```