

Solution to Exercises in L#15

Solution to Exercise (1) on Slide 14

- Find errors, if any, in the following function prototypes

- `int func1(int a, void);`

`void cannot be combined with other types! To fix the error: int func1(void); or int func1(int a);`

- `float func2 (int a; int b)`

`change ; to , and ; is missing at the end → float func2(int a, int b);`

- `void func3(int, double, int);`

`No error`

- `int func4(a,b,c);`

`parameter types are necessary, but parameter names are not! E.g., int func4(int, int, int);`

- `void func5(int n, double, int m);`

`no error but not a good form, parameter names should be either all given or not given!`

- `void func6(int,a, int,b, float,c);`

`no , between type and name → void func6(int a, int b, float c);`

Solution to Exercise (2) on Slide 18

- What is the output from this program

```
#include <stdio.h>
void NumOut(int m, float n);
void main (void)
{
    int a=1, b=2, c=3;
    float r=1.32, s=3.23, t=0.32;
    NumOut (a,b);
    NumOut (r,s);
}
void NumOut (int m, float n)
{
    printf("m=%d, n=%f\n", m, n);
}
```

m=1, n=2.000000

m=1, n=3.230000

Solution to Exercise (3) on Slide 19

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

```
/*function declaration*/
```

```
int func_sum(int, int) → error 1: semicolon is missing  
at the end
```

```
void main (void); → error 2: no semicolon  
{ in function header
```

```
int a=3, b=7, c=6;  
float u=6.372, v=1.23;
```

```
/*function calls*/
```

```
printf(“%d\n”, func_sum1(a,b)); → error 3: function name should be func_sum  
printf(“%d\n”, func_sum2(a,u)); → error 4: function name should be func_sum  
printf(“%d\n”, func_sum3(b,c,u)); → errors 5 & 6: function name should be  
func_sum and the number of parameters  
should be 2 (e.g., remove ,u)
```

```
/*function definition*/
```

```
void func_sum(int m, int n); → error 7: no ; at the end of function header
```

```
{  
    return m+n; → error 8: what you are returning from the function  
} should match the return type of the function → change  
void in the function header to int!
```

Output:

10

9

13