

# Function Overview (1B)

Copyright (c) 2009-2016 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".

## f1.c

```
#include <stdio.h>

void func(int x) {
    printf("&x= %p x= %d \n", &x, x);
    x *= 2;
    printf("&x= %p x= %d \n", &x, x);
}

void main(void) {
    int a = 100;

    printf("&a= %p a= %d \n", &a, a);

    func( a );

    printf("&a= %p a= %d \n", &a, a);
}
```

```
&a= 0xbf94b1dc a= 100
&x= 0xbf94b1c0 x= 100
&x= 0xbf94b1c0 x= 200
&a= 0xbf94b1dc a= 100
```

## f2.c

```
#include <stdio.h>

int func(int x) {
    printf("&x= %p x= %d \n", &x, x);
    x *= 2;
    printf("&x= %p x= %d \n", &x, x);
    return x;
}

void main(void) {
    int a = 100;

    printf("&a= %p a= %d \n", &a, a);

    a = func( a );

    printf("&a= %p a= %d \n", &a, a);
}
```

```
&a= 0xbfff505c a= 100
&x= 0xbfff5040 x= 100
&x= 0xbfff5040 x= 200
&a= 0xbfff505c a= 200
```

### f3.c

```
#include <stdio.h>

void func(int *x) {
    printf("&x= %p x= %p *x= %d \n", &x, x, *x);
    *x *= 2;
    printf("&x= %p x= %p *x= %d \n", &x, x, *x);
}

void main(void) {
    int a = 100;

    printf("&a= %p a= %d \n", &a, a);

    func( &a );

    printf("&a= %p a= %d \n", &a, a);
}
```

```
&a= 0xbfa7d7fc a= 100
&x= 0xbfa7d7e0 x= 0xbfa7d7fc *x= 100
&x= 0xbfa7d7e0 x= 0xbfa7d7fc *x= 200
&a= 0xbfa7d7fc a= 200
```

#### f4.c

```
#include <stdio.h>

void exchange(int *x, int *y) {
    int t;

    printf("&x= %p x= %p *x= %d \n", &x, x, *x);
    printf("&y= %p y= %p *y= %d \n", &y, y, *y);

    t = *x;
    *x = *y;
    *y = t;

    printf("&x= %p x= %p *x= %d \n", &x, x, *x);
    printf("&y= %p y= %p *y= %d \n", &y, y, *y);
}

void main(void) {
    int a = 100;
    int b = 200;

    printf("&a= %p a= %d \n", &a, a);
    printf("&b= %p b= %d \n", &b, b);

    exchange( &a, &b );

    printf("&a= %p a= %d \n", &a, a);
    printf("&b= %p b= %d \n", &b, b);
}
```

```
&a= 0xbff6fecc a= 100
&b= 0xbff6fec8 b= 200
&x= 0xbff6feb0 x= 0xbff6fecc *x= 100
&y= 0xbff6feb4 y= 0xbff6fec8 *y= 200
&x= 0xbff6feb0 x= 0xbff6fecc *x= 200
&y= 0xbff6feb4 y= 0xbff6fec8 *y= 100
&a= 0xbff6fecc a= 200
&b= 0xbff6fec8 b= 100
```

**t1.c**

```
#include <stdio.h>

int main(void)
{
    int S1, S2, S3;

    { int n=1;      // parameter var n : input
      int k, S=0;
      for (k=1; k<=n; ++k) S += k;
      S1 = S;      // return var S : output
    }
    printf("psum(1) = %d \n", S1);

    { int n=2;      // parameter var n : input
      int k, S=0;
      for (k=1; k<=n; ++k) S += k;
      S2 = S;      // return var S : output
    }
    printf("psum(2) = %d \n", S2);

    { int n=3;      // parameter var n : input
      int k, S=0;
      for (k=1; k<=n; ++k) S += k;
      S3 = S;      // return var S : output
    }
    printf("psum(3) = %d \n", S3);

}
```

**a1.c**

```
// a1.c
#include <stdio.h>

int psum (int n) // parameter var n : input
{
    int k, S=0;    // local variable
    for (k=1; k<=n; ++k) S += k;
    return S;     // return var S : output
}

int main(void)
{
    int S1, S2, S3;

    S1 = psum( 1 );
    printf("psum(1) = %d \n", S1);

    S2 = psum( 2 );
    printf("psum(2) = %d \n", S2);

    S3 = psum( 3 );
    printf("psum(3) = %d \n", S3);
}
```

## A2.c

```
// a2.c
#include <stdio.h>

int psum (int n); // psum prototype

int main(void)
{
    int S1, S2, S3;

    S1 = psum( 1 );
    printf("psum(1) = %d \n", S1);

    S2 = psum( 2 );
    printf("psum(2) = %d \n", S2);

    S3 = psum( 3 );
    printf("psum(3) = %d \n", S3);
}

int psum (int n) // psum definition
{
    int k, S=0;
    for (k=1; k<=n; ++k) S += k;
    return S;
}
```



### a3.c

```
// a3.c main only
#include <stdio.h>

int psum (int n); // psum prototype

int main(void)
{
    int S1, S2, S3;

    S1 = psum( 1 );
    printf("psum(1) = %d \n", S1);

    S2 = psum( 2 );
    printf("psum(2) = %d \n", S2);

    S3 = psum( 3 );
    printf("psum(3) = %d \n", S3);
}
```

**a4.c**

```
// a4.c  psum only

int psum (int n) // parameter var n : input
{
    int k, S=0;    // local variable
    for (k=1; k<=n; ++k) S += k;
    return S;      // return var S : output
}
```

## a5.c

```
// a5.c  main only
// a5.h  header file : psum prototype
#include <stdio.h>
#include "a5.h"

int main(void)
{
    int S1, S2, S3;

    S1 = psum( 1 );
    printf("psum(1) = %d \n", S1);

    S2 = psum( 2 );
    printf("psum(2) = %d \n", S2);

    S3 = psum( 3 );
    printf("psum(3) = %d \n", S3);
}
```

**5.h**

```
int psum( int );
```