Passing function parameter as a pointer

Parameter Passing by Pointer

- A function may return multiple values by declaring their formal parameters (passing value) as pointers variables.
- This way of passing the argument is known as call by reference
- When the value referenced by the pointer is changed inside the function, the value in the actual variable will also change.
- Therefore, we can pass the result of the function through the function argument without having to use the *return* statement.

Parameter Passing by Pointer

- When a pointer is passed to a function, we are actually passing the address of a variable to the function.
- Since we have the address, we can directly manipulate the data in the address.
- In the case where a non-pointer variable is passed, the function will create another space in memory to hold the value locally while the program is inside the function. Therefore, any change to the variable inside the function will not change the actual value of the variable.

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(x,y);//a = x; b = y;
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(x,y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:

Before swap function: x = 5, y = 10

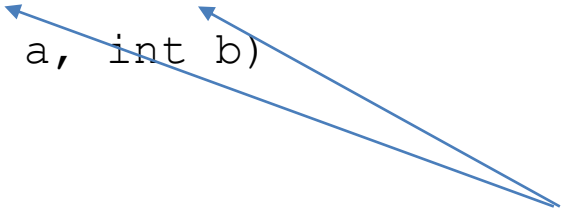
After swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(x,y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Local variables (gets destroyed after function ends, no effect on x and y inside main)

Output:

Before swap function: x = 5, y = 10

After swap function: x = 5, y = 10

Parameter Passing by Pointer

- Declare the parameters of `swap` as pointer variables so that they can contain addresses.

```
void swap(int *addr1, int * addr2)
```

- We will place the addresses of `x` and `y` into `addr1` and `addr2`, respectively.

```
swap(&x, &y);
```

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
    int temp;
    temp = *addr1;
    *addr1 = *addr2;
    *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);//addr1=&x; addr2=&y;
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
    int temp;
    temp = *addr1;
    *addr1 = *addr2;
    *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:

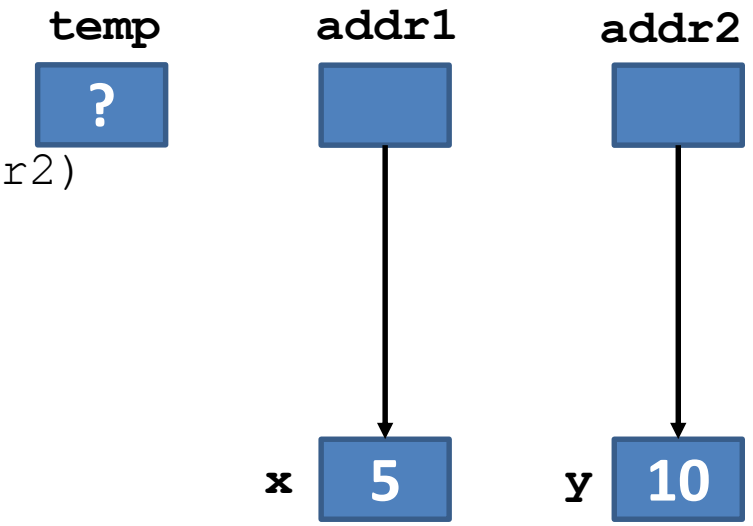
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Output:

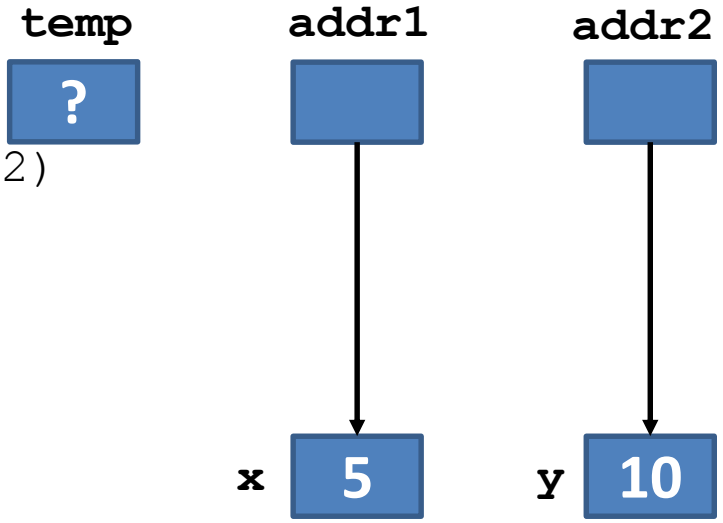
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Output:

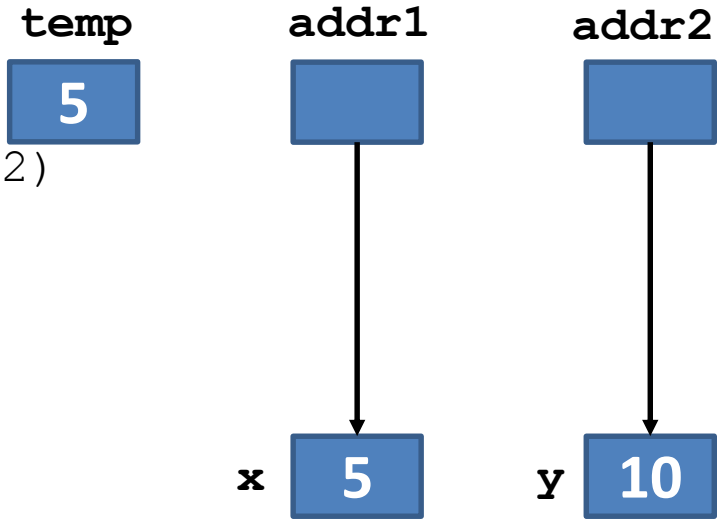
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Output:

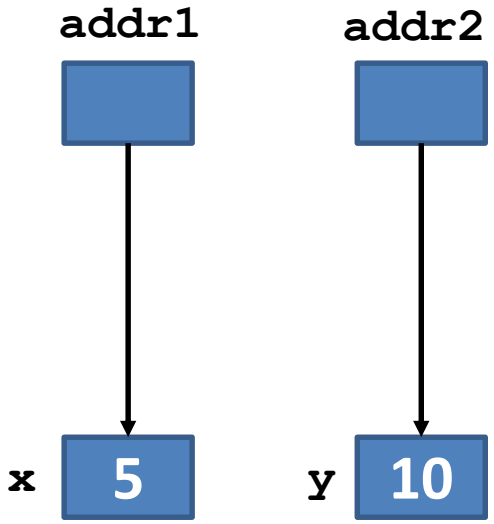
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Output:

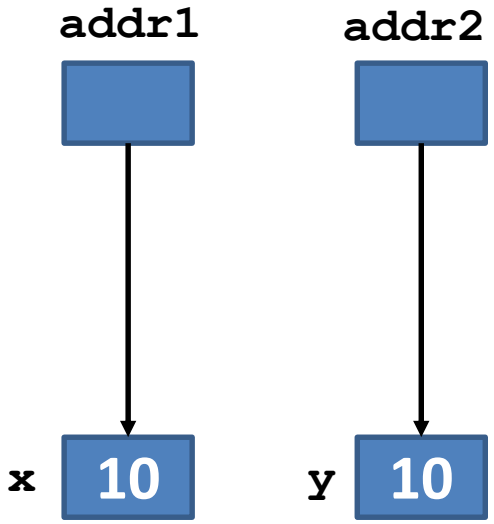
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Output:

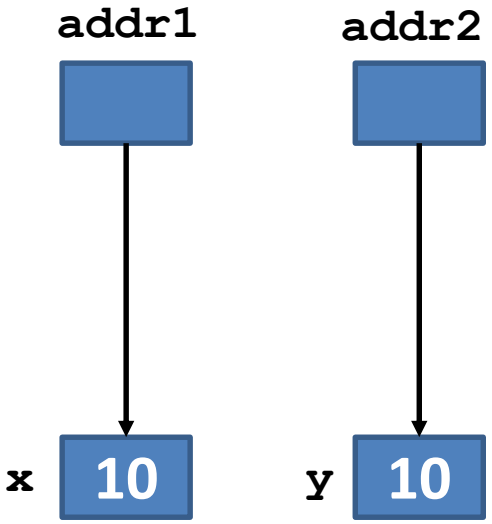
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
    int temp;
    temp = *addr1;
    *addr1 = *addr2;
    *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Output:

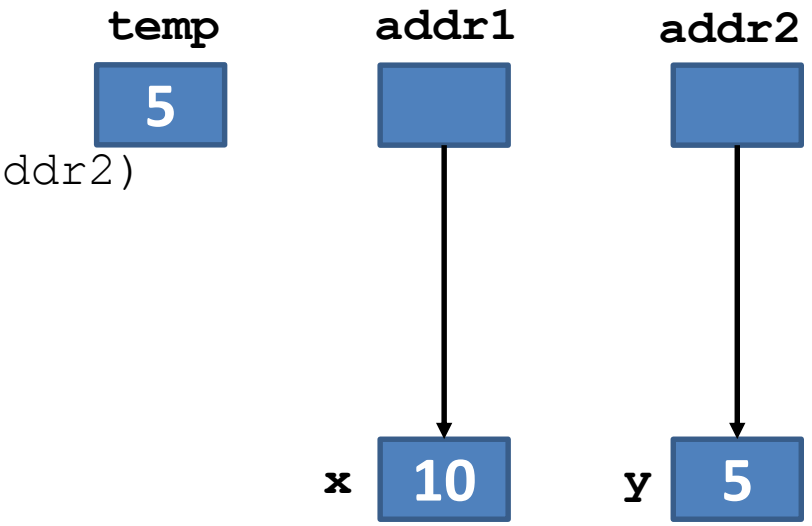
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
    int temp;
    temp = *addr1;
    *addr1 = *addr2;
    *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```



Output:

Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:
Before swap function: x = 5, y = 10

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:
Before swap function: x = 5, y = 10
After swap function: x = 10, y = 5

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
     int temp;
     temp = *addr1;
     *addr1 = *addr2;
     *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:
Before swap function: x = 5, y = 10
After swap function: x = 10, y = 5

Parameter Passing by Pointer

```
#include <stdio.h>

void swap(int *addr1, int *addr2)
{
    int temp;
    temp = *addr1;
    *addr1 = *addr2;
    *addr2 = temp;
}

int main(void)
{
   int x = 5, y = 10;
   printf("Before swap function: x = %d, y = %d\n",x,y);
   swap(&x,&y);
   printf("After swap function: x = %d, y = %d",x,y);
   return 0;
}
```

Output:

Before swap function: x = 5, y = 10

After swap function: x = 10, y = 5