함 수 화 1
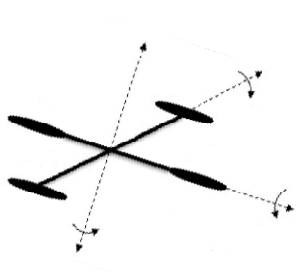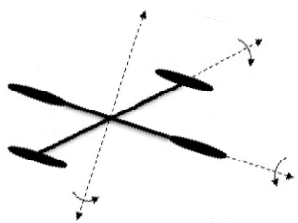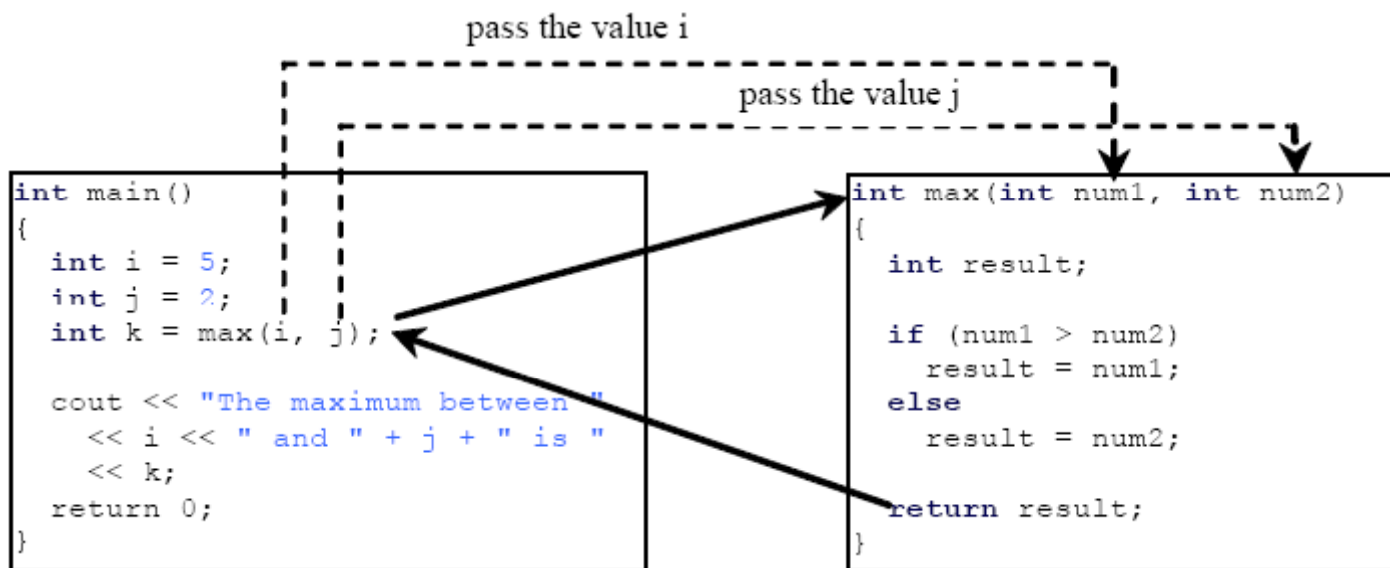
- 함수 function

```
int z = max(x, y);
        ↑     ↑
     actual parameters
       (arguments)
```
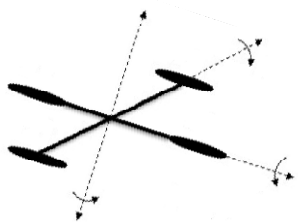
```
                    return value type    method name    formal parameters
                              ↓               ↓                ↓
function     →    int max (int num1, int num2)
header
                  {                                    └──────────────┘
                                                              ↑
function     →      int result;                        parameter list
body
                    if (num1 > num2)
                      result = num1;                return value
                    else
                      result = num2;                   ↙
                                                    ↙
                    return result;
                  }
```

```
pass the value i
pass the value j

int main()                          int max(int num1, int num2)
{                                   {
  int i = 5;                          int result;
  int j = 2;
  int k = max(i, j);                  if (num1 > num2)
                                        result = num1;
  cout << "The maximum between "      else
    << i << " and " + j + " is "        result = num2;
    << k;
  return 0;                           return result;
}                                   }
```

함수의 반환값이 없을 때는 void로 선언

```cpp
#include <iostream>
using namespace std;

void swap(int n1, int n2)
{
  cout << "\t Inside the swap function" << endl;
  cout << "\t\t Before swapping n1 is " << n1 << " n2 is " << n2 << endl;

  // Swap n1 with n2
  int temp = n1;
  n1 = n2;
  n2 = temp;

  cout << "\t\t After swapping n1 is " << n1 << " n2 is " << n2 << endl;
}

int main()
{
  // Declare and initialize variables
  int num1 = 1;
  int num2 = 2;

  cout << "Before invoking the swap function, num1 is "<< num1 << " and num2 is " << num2 << endl;

  // Invoke the swap function to attempt to swap two variables
  swap(num1, num2);

  cout << "After invoking the swap function, num1 is " << num1 << " and num2 is " << num2 << endl;

        system("pause");
  return 0;
}
```
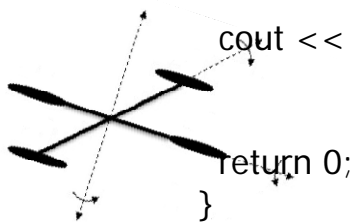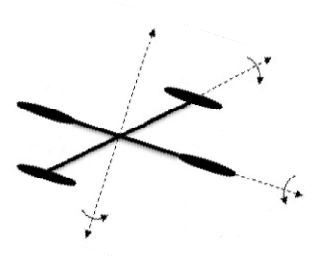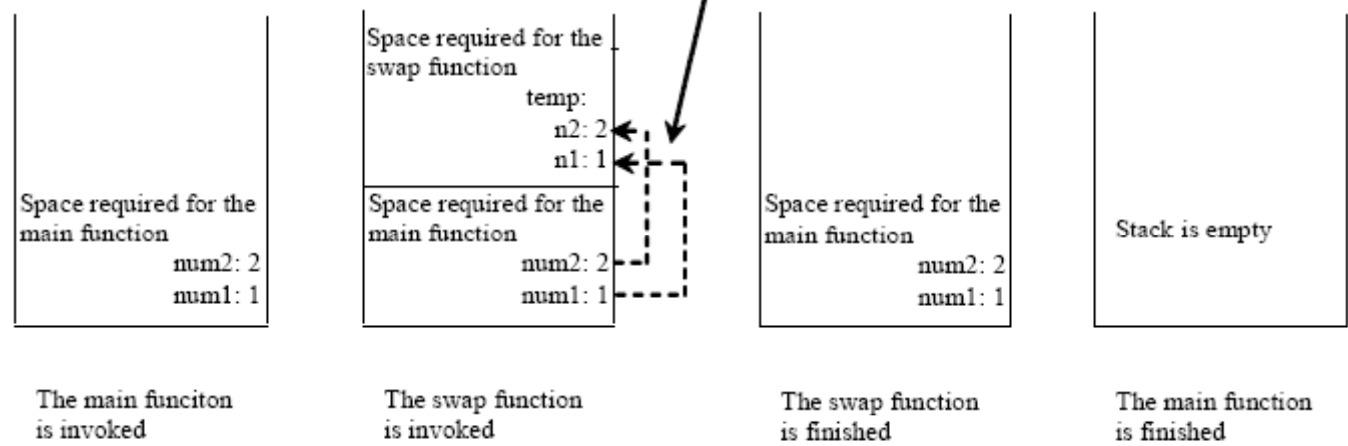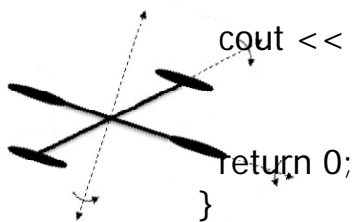
Before invoking the swap function, num1 is 1 and num2 is 2
        Inside the swap function
                Before swapping n1 is 1 n2 is 2
                After swapping n1 is 2 n2 is 1
After invoking the swap function, num1 is 1 and num2 is 2
계속하려면 아무 키나 누르십시오 . . .

The values of num1 and num2 are
passed to n1 and n2. Executing swap
does not affect num1 and num2.

Space required for the
swap function
        temp:
        n2: 2
        n1: 1

Space required for the main function	Space required for the main function	Space required for the main function	Stack is empty
num2: 2	num2: 2	num2: 2	
num1: 1	num1: 1	num1: 1	

The main funciton   The swap function   The swap function   The main function
is invoked   is invoked   is finished   is finished

Pink Wink

```cpp
#include <iostream>
using namespace std;

void swap(int &n1, int &n2)
{
  cout << "\t Inside the swap function" << endl;
  cout << "\t\t Before swapping n1 is " << n1 << " n2 is " << n2 << endl;

  // Swap n1 with n2
  int temp = n1;
  n1 = n2;
  n2 = temp;

  cout << "\t\t After swapping n1 is " << n1 << " n2 is " << n2 << endl;
}

int main()
{
  // Declare and initialize variables
  int num1 = 1;
  int num2 = 2;

  cout << "Before invoking the swap function, num1 is "<< num1 << " and num2 is " << num2 << endl;

  // Invoke the swap function to attempt to swap two variables
  swap(num1, num2);

  cout << "After invoking the swap function, num1 is " << num1 << " and num2 is " << num2 << endl;

          system("pause");
  return 0;
}
```
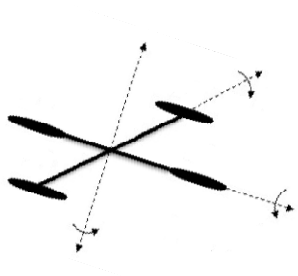
```
C:\Documents and Settings\Administrator\바탕 화면\Project1.exe

Before invoking the swap function, num1 is 1 and num2 is 2
        Inside the swap function
                Before swapping n1 is 1 n2 is 2
                After swapping n1 is 2 n2 is 1
After invoking the swap function, num1 is 2 and num2 is 1
계속하려면 아무 키나 누르십시오 . . . _
```

참조 변수 (reference variable)은 원 변수를 참조하기 위해 함수
의 매개변수로 사용되는 변수. 참조변수를 수정하면 원변수도 수
정됨.

- 함수의 오버로드(Overload)

```cpp
#include <iostream>
using namespace std;

/** Return the max between two int values */
int max(int num1, int num2)
{
  if (num1 > num2)
    return num1;
  else
    return num2;
}

/** Find the max between two double values */
double max(double num1, double num2)
{
  if (num1 > num2)
    return num1;
  else
    return num2;
}

/** Return the max among three double values */
double max(double num1, double num2, double num3)
{
  return max(max(num1, num2), num3);
}
```
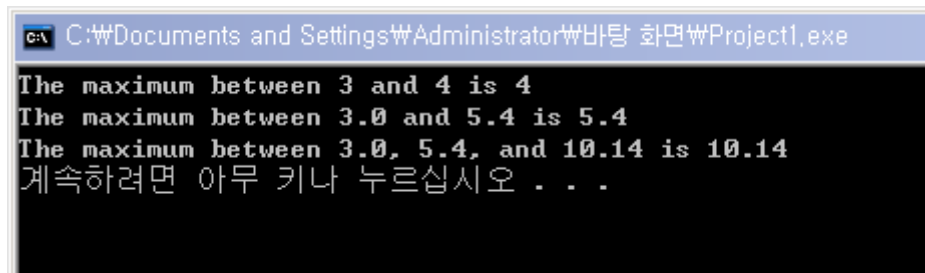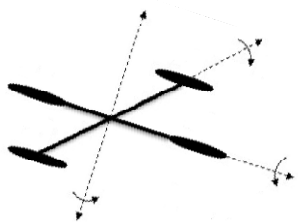
```cpp
int main()
{
  // Invoke the max function with int parameters
  cout << "The maximum between 3 and 4 is " << max(3, 4) << endl;

  // Invoke the max function with the double parameters
  cout << "The maximum between 3.0 and 5.4 is " << max(3.0, 5.4) << endl;

  // Invoke the max function with three double parameters
  cout << "The maximum between 3.0, 5.4, and 10.14 is " << max(3.0, 5.4, 10.14) << endl;

  system("pause");
  return 0;
}
```
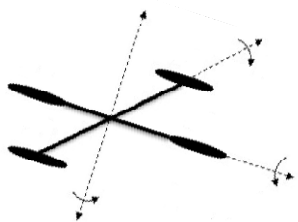
```
C:\Documents and Settings\Administrator\바탕 화면\Project1.exe
The maximum between 3 and 4 is 4
The maximum between 3.0 and 5.4 is 5.4
The maximum between 3.0, 5.4, and 10.14 is 10.14
계속하려면 아무 키나 누르십시오 . . .
```

```cpp
#include <iostream>
using namespace std;

// Function prototype
int max(int num1, int num2);
double max(double num1, double num2);
double max(double num1, double num2, double num3);

int main()
{
    cout << "The maximum between 3 and 4 is " << max(3, 4) << endl;
    cout << "The maximum between 3.0 and 5.4 is " << max(3.0, 5.4) << endl;
    cout << "The maximum between 3.0, 5.4, and 10.14 is " << max(3.0, 5.4, 10.14) << endl;

    return 0;
}

int max(int num1, int num2)
{
    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
}

double max(double num1, double num2)
{
    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
}

double max(double num1, double num2, double num3)
{
    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
}
```