20164389 멀티미디어공학과 박

김상엽 지음 [이지스 퍼블리

```
void main ( )
{
      Add();
}
```

Add 함수의 기계어
Sub 함수의 기계어

main 함수의 기계어

main 함수의 기계어
Add 함수의 기계어
Sub 함수의 기계어

```
void main ( )
{
      Add();
}
```

Add 함수의 기계어
Sub 함수의 기계어

main 함수의 기계어

main 함수의 기계어
Add 함수의 기계어

```
void main ( )
{
        int result1, result2, result3, result4;
        result1 = Add(2, 3); /*오류발생 */
        result2 = Sub(2, 3); /*오류발생 */
        result3 = Mul(2, 3); /*오류발생 */
        result4 = Div(2, 3);  /*오류발생 */
}
```

Add, Sub, Mul, Div 함수가 있지만 기계어로 번역되어 내용을 확인할 수 없음

(MyMath.lib 파일에 있는 함수를 사용하기 위해 각 함수 원형을 선언함)

```
int Add(int value1, int value2);
int Sub(int value1, int value2);
int Mul(int value1, int value2);
int Div(int value1, int value2);
```

Add, Sub, Mul, Div 함수가 있지만 기계어로 번역되어 내용을 확인할 수 없음

```
#include <stdio.h>
#include “Mymeth.h”
```

```
#include “MyMath.h”
#include “C:₩download₩MyMath.h”
```

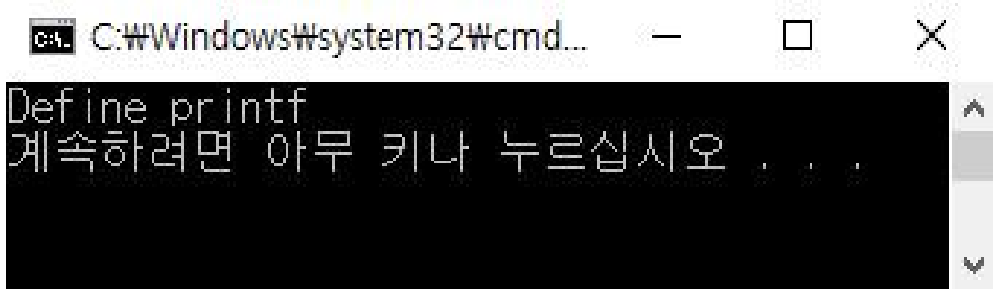
```
#include "MyMath.h"
void main ( )
{
        int result1, result2, result3, result4;
        result1 = Add(2, 3); /*오류발생 */
        result2 = Sub(2, 3); /*오류발생 */
        result3 = Mul(2, 3); /*오류발생 */
        result4 = Div(2, 3);  /*오류발생 */
}
```

```
(MyMath.lib 파일에 있는 함수를 사용하기
위해 각 함수 원형을 선언함)
int Add(int value1, int value2);
int Sub(int value1, int value2);
int Mul(int value1, int value2);
int Div(int value1, int value2);
```

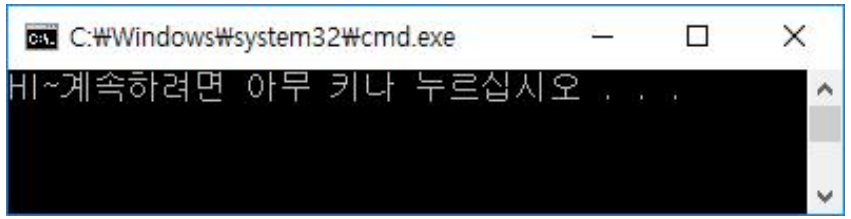
Add, Sub, Mul, Div 함수가
있지만 기계어로 번역되어
내용을 확인할 수 없음

```
#define MAX_COUNT 3

#define POW_VALUE(a) (a * a)
int data = POW_VALUE(3);
```

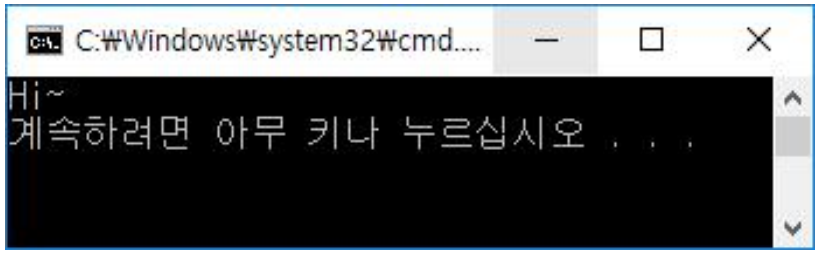
```c
#include <stdio.h>
#define PRINT printf("Define printf ₩n")

int main()
{
    PRINT;
    return 0;
}
```



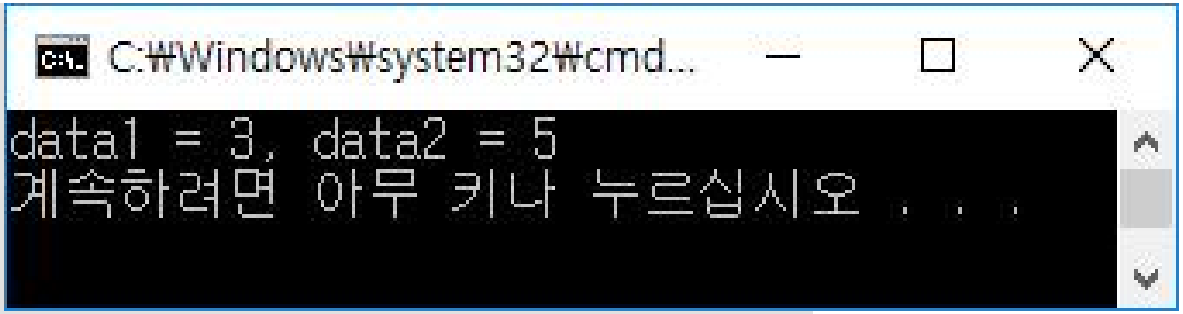
```c
#include <stdio.h>

void main()
{
    putchar('H');
    putchar('I');
    putchar('~');
}
```



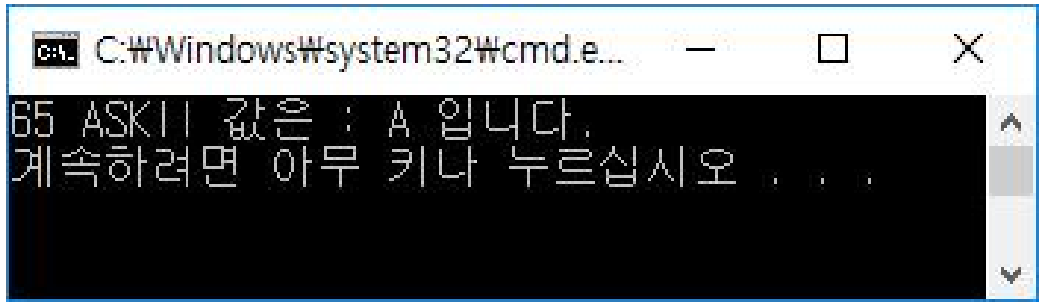

```c
#include <stdio.h>

void main()
{
    puts("Hi~");
}
```



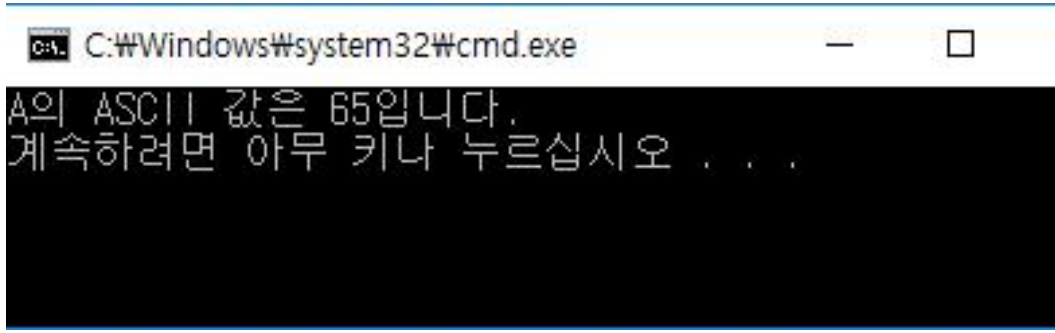
```c
#include <stdio.h>

void main()
{
    int data1 = 3;
    int data2 = 5;

    printf("data1 = %d, data2 = %d ₩n", data1, data2);
}
```



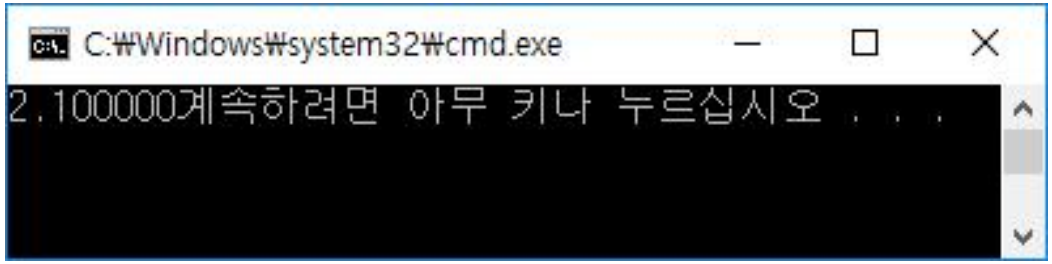
```c
#include <stdio.h>

void main()
{
    printf("65 ASKII 값은 : %c 입니다. ₩n", 65);
}
```



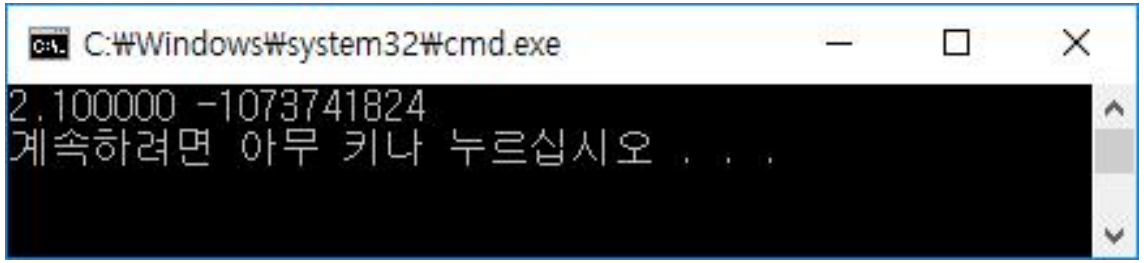

```c
#include <stdio.h>

void main()
{
    char data = 65;
    printf("%c의 ASCII 값은 %d입니다.₩n", data, data);
}
```



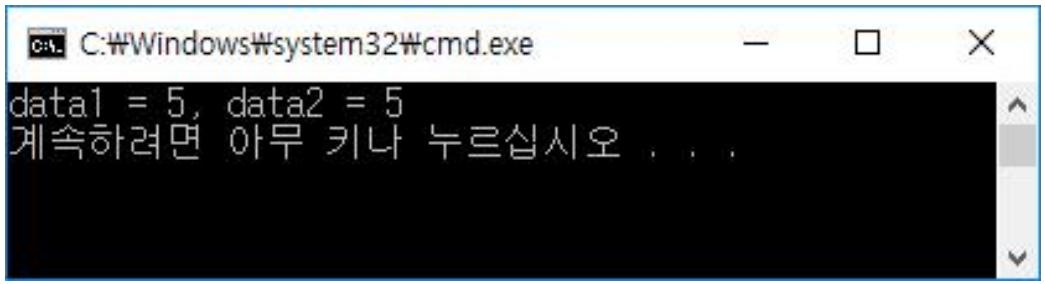
```c
#include <stdio.h>

void main()
{
    float value = 2.1f;
    printf("%f", value);
}
```



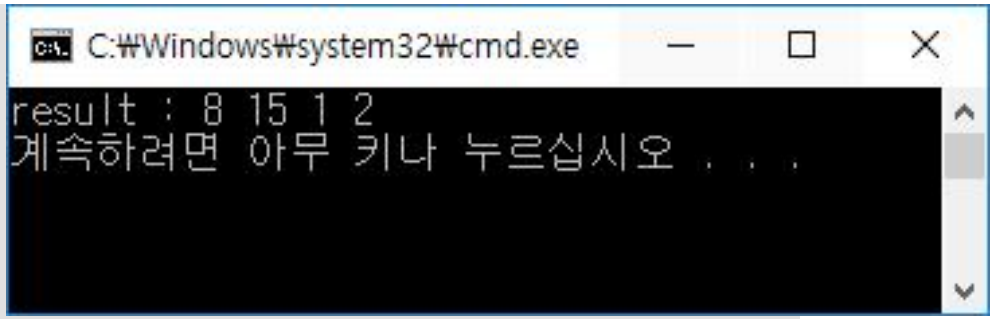

```c
#include <stdio.h>

void main()
{
    float value = 2.1f;
    printf("%f %d ₩n", value, value);
}
```



```c
#include <stdio.h>

void main()
{
    int data1, data2;
    data1 = 5;
    data2 = data1;
    printf("data1 = %d, data2 = %d ₩n", data1, data2);
}
```



```c
void main()
{
    int data1 = 5, data2 = 3;

    int result1 = data1 + data2;
    int result2= data1 * data2;
    int result3 = data1 / data2;
    int result4 = data1 % data2;

    printf("result : %d %d %d %d ₩n", result1, result2, result3, result4);
}
```



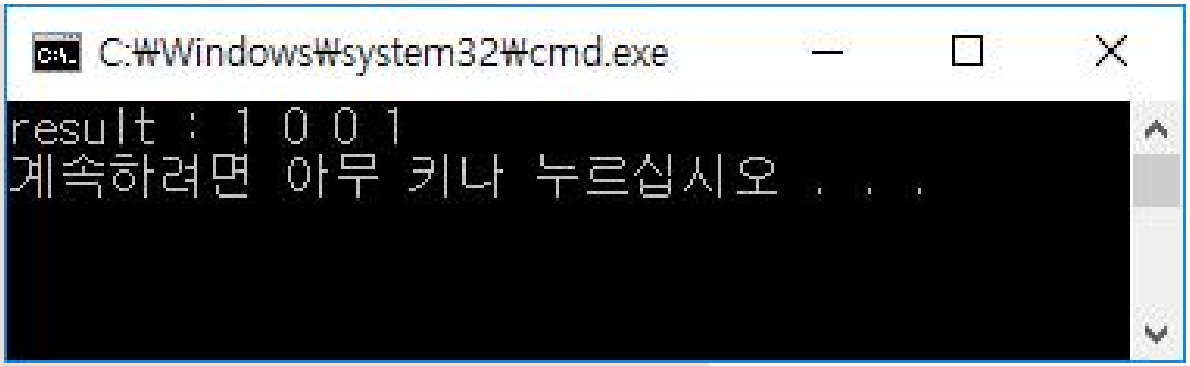
전위형

```
int i = 5, sum;
sum = ++i;
```

후위형

```
int i = 5, sum;
sum = i++;
```

```c
#include <stdio.h>

void main()
{
    int data1 = 5, data2 = 3;

    int result1 = data1 > data2;
    int result2= data1 <= data2;
    int result3 = data1 == data2;
    int result4 = data1 != data2;

    printf("result : %d %d %d %d ₩n", result1, result2, result3, result4);
}
```



A < B	A가 B보다 작으면 참	A > B	A가 B보다 크면 참
A <= B	A가 B보다 작거나 같으면 참	A >= B	A가 B보다 크거나 같으면 참
A == B	A와 B가 같으면 참	A != B	A와 B가 같지 않으면 참

0	0	0	0	1
0	1	0	1	
1	0	0	1	0
1	1	1	1	

논리 연산자	연산 특성
&&	A와 B가 모두 참이어야 결과 값이 참, 하나라도 거짓이면 거짓
||	A 또는 B가 참이면 참, 둘 다 거짓일 때만 거짓
!	A가 거짓이면 참, 참이면 거짓

```c
#include <stdio.h>

void main()
{
    int data1 = 5, data2 = 3;
    int result1 = 0 || 1;
    int result2 = 3 && -1;
    int result3 = data1 == 3 || data2 == 3;
    int result4 = data1 == 3 && data2 == 3;
    int result5 =! data1;

    printf("result : %d %d %d %d %d₩n", result1, result2, result3, result4, result5);
}
```