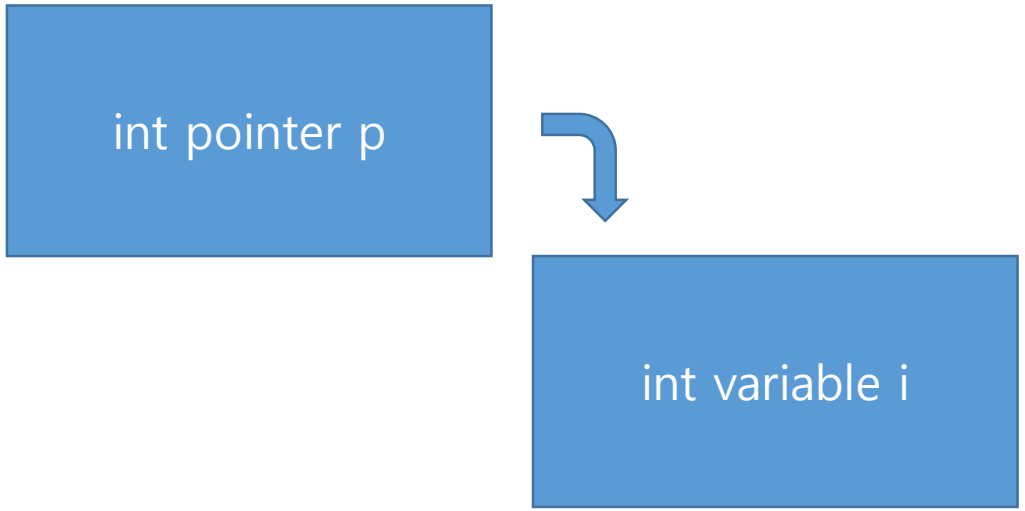
C Programming

Practice 9

Pointer Declaration and Assignment

1. Size of pointer = 4 bytes
2. Save the address value

```
int i, *p;  /* p is of type "pointer to int" */

p = &i;
p = NULL;
```



Array

- Define array

```
int grade [3];
int grade [3] = {0};
char characters[3];
```


```
int a[] = {3, 4, 5};
```

A string

- A string is terminated by the end-of-string sentinel ₩0, or null character.

- Consumes one character storage. E.g., a constant string “abc” is stored in memory as four characters, the last one being ₩0.

Reading in a string

```
#define  MAXWORD  100
int main(void)
{
    char  w[MAXWORD];
    ......
```

- character by character:

```
w[0] = 'A';  w[1] = 'B';  w[2] = 'C';  w[3] = '₩0';
```

- by use of scanf():

```
scanf("%s", w);
```

Initialization of Strings

- char w[] = {‘a’, ‘b’, ‘c’, ‘₩0’};
- char w[] = “abc”;
- char *w = “abc”;
- printf("%s %s₩n", w, w+1); /* abc bc is printed */

nice_day.c

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

 #define   MAXSTRING   100

int main(void)
{
    char   c, name[MAXSTRING];
    int    i, sum = 0;

    printf("₩nHi!  What is your name?  ");
    for (i = 0; (c = getchar()) != '₩n'; ++i) {
        name[i] = c;
        if (isalpha(c))       /* sum the letters */
            sum += c;
    }

    name[i] = '₩0';
    printf("₩n%s%s%s₩n%s",
        "Nice to meet you ", name, ".",
        "Your name spelled backwards is ");
    for (--i; i >= 0; --i)
        putchar(name[i]);
    printf("₩n%s%d%s₩n₩n%s₩n₩n",
        "and the letters in your name sum to ",
        sum, ".", "Have a nice day!");
    return 0;
}
```

```
Hi! What is your name?  C. B. Diligent

Nice to meet you C. B. Diligent.
Your name spelled backward is tnegiliD .B .C
and the letters in your name sum to 949.

Have a nice day!
```

Passing arguments to main()

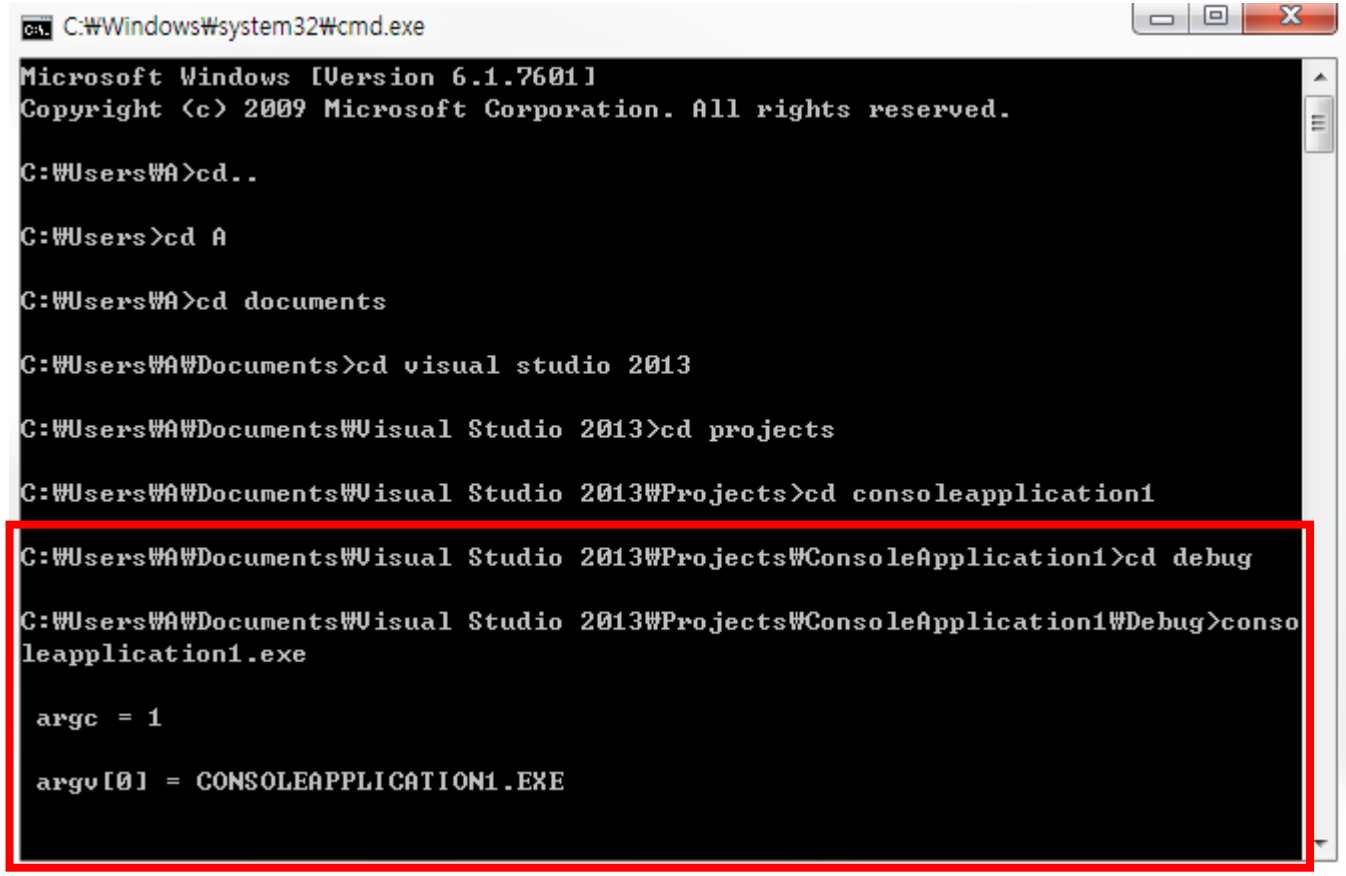
- argc: a count of the command line arguments.
- argv: an array of pointers to char; an array of strings.

```
int main( int argc, char *argv[])
```

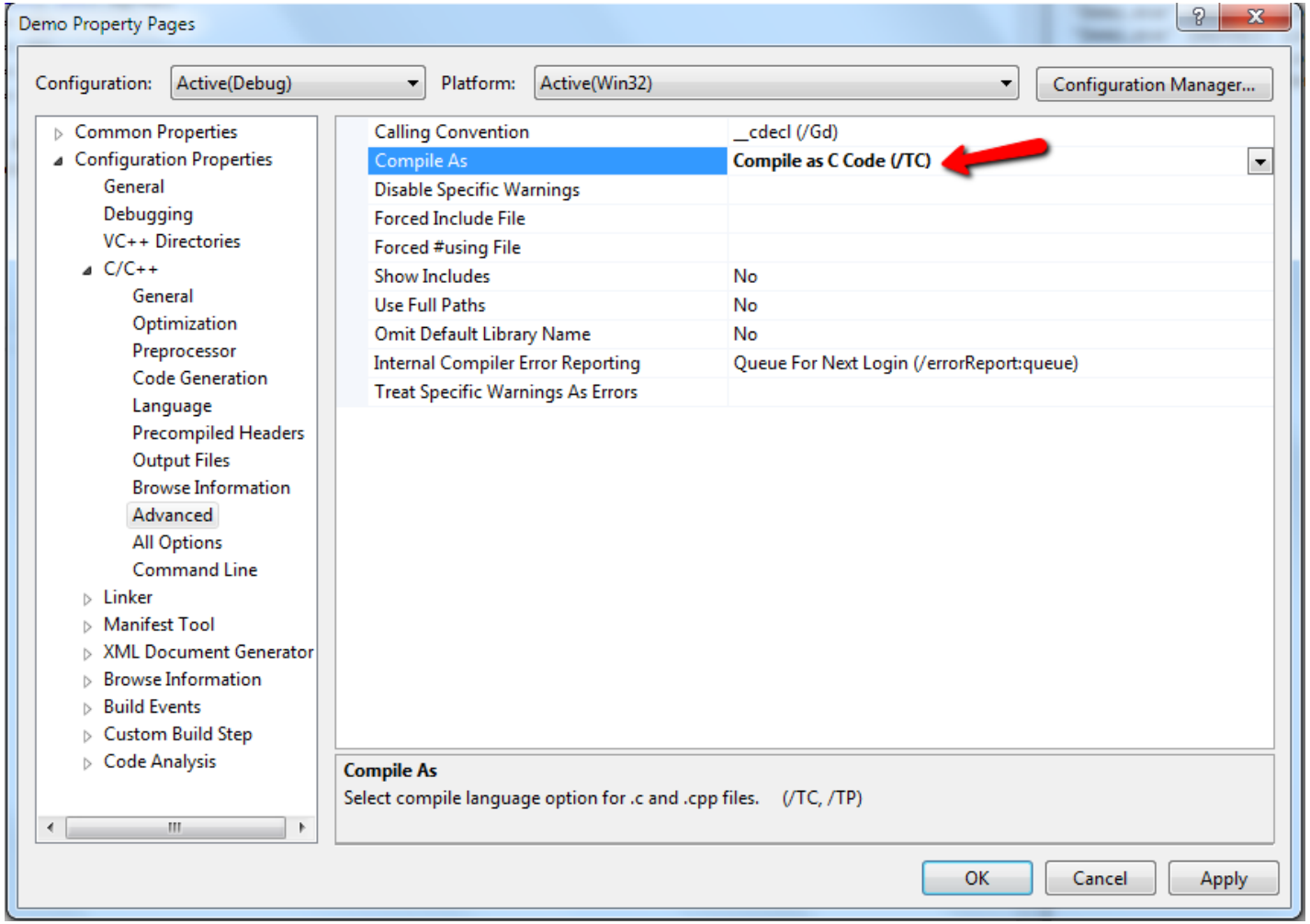
echo.c

```c
#include <ctype.h>
#include <stdio.h>
#include <String.h>
const int   maxstring = 100;

int main(int argc, char *argv[])
{
    char   copy[maxstring];
    char   *p;
    int    i;

    printf("₩n   argc = %d₩n₩n", argc);
    for (i = 0; i < argc; ++i) {
        strcpy(copy, argv[i]);
        for (p = copy; *p != '₩0'; ++p)
            *p = toupper(*p);
        printf("   argv[%d] = %s₩n", i, copy);
    }
    putchar('₩n');
    return 0;
}
```

argc, argv program debugging in commend line



argc, argv program debugging in Visual studio 2010



String.h

- char *strcat(char *s1, const char *s2);
- int strcmp(const char *s1, const char *s2);
- int strncmp(const char *s1, const char *s2, size_t num);
- char *strcpy(char *s1, const char *s2);
- unsigned strlen(const char *s);

Homework18 – Compare string1 and string2

- Input the number to check
- Input the string1
- Input the string2
- Compare the string1 and string2

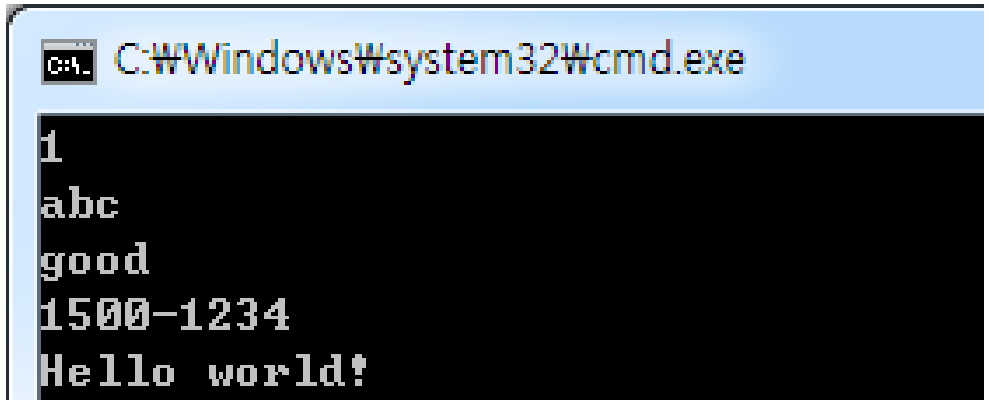
```
Input the number to check : 4
Input the string1 : abcd1234
Input the string2 : abcdefgh
equal 4 character
```

```
Input the number to check : 5
Input the string1 : abc
Input the string2 : abc
equal
```

```
Input the number to check : 5
Input the string1 : a
Input the string2 : b
not equal
```

Homework19 – String 1~5 sorting

- char string1[20] = "abc";
- char string2[20] = "1";
- char string3[20] = "Hello world!";
- char string4[20] = "good";
- char string5[20] = "1500-1234";



Homework form

- Homework submission e-mail:

hizorro99@naver.com

- E-mail title: day(Thursday or Friday)_name_#week
  - Ex) Friday_james_week11
  - Ex) 목요일반_장원철_11주차
- File title: student id_name_#.c
  - Ex) 2014123456_james_18.c (or .cpp)
  - Ex) 2014123456_james_19.c (or .cpp)