Homework #6

[ECE10002] C Programming

Mission

- Solve Problem 1, 2, 3 in C language.
- Submit through Hisnet
  - Due date: PM11:00, Dec. 7th (Sat.)
  - Submit a zip file including three files: hw6_1.c, hw6_2.c, hw6_3.c
  - Use the skeleton codes for Problem 2, and 3.
- All variables, except for array variables, should be initialized properly.

  Ex) int i = 0, *p = NULL; // Okay!

  int i, *p; // Uninitialized variables can be penalized
- Every program should contain algorithm in pseudo code.
  - Use comments to describe each step of the algorithm

Problem 1

- Write a program that reads a series of integers numbers from the user, and then print them in reverse order.
  - When it starts, it should ask the number of data, *n*.
    - *n* is unlimited.

  ➔ Dynamically allocate an array of size *n*.
  - You MUST check if memory allocation was successful.

Ex) hw6_1.exe
How many integers? *3*
Input integer numbers.
10
30
40
Numbers in reverse order: 40 30 10

Problem 2

- Implement two functions ExtendString() and ShiftString()
  - void ExtendString(char str[], int newLen)
    - Appends ' 's at the end of str to make its length newLen

    Ex) char string[21] = "Hello";
    ExtendString(string, 20);
    printf("string = [%s]₩n", string);

    Result)
    string = [Hello]

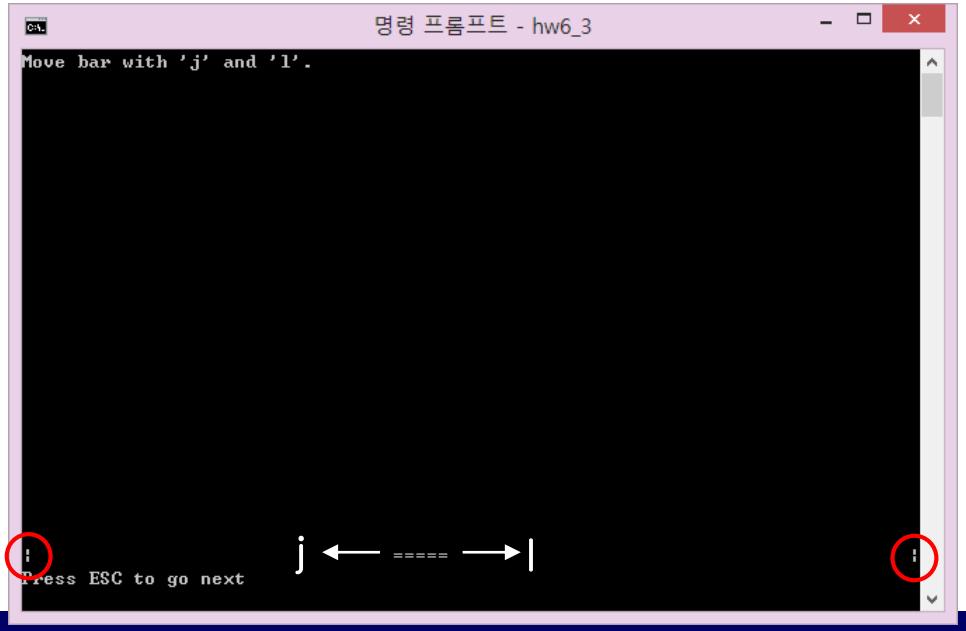
  - void ShiftString(char str[])
    - Shifts all the characters in *str* one step right
    - The last character should move to *str[0]*.

    Ex) char string[21] = "Hello";
    ShiftString(string);
    printf("string = [%s]₩n", string);
    getchar();

    Result) string = [oHell]

Problem 3

- Write a function that displays a moving bar
  - length of the bar: 5
  - j: start left move, l: start right move, other key: stop
    - If ‘j’ or ‘l’ was pressed, the bar should keep moving until ’k’ is pressed or the bar hits the boundary.
  - Boundary (‘|’ sign)
    - leftBound = 1
    - rightBound = 80
  - Acceleration: optional
    - [−3,+3]



Non-Blocking Key Input

- For this problem, you must use non-blocking key input
  - See “Console Interface.pdf”
- Use if(kbhit()) before call getch()

Ex)

```
void NonblockingInput()
{
    int i = 1;
    char c = 0;

    while(c != 't'){
       printf("%d ", i++);
       if(kbhit())          // check if a key is pressed
            c = getch();
    }
}
```

Console Interface Functions

- Function to clear the screen

```
void clrscr(void)              // clear the screen
{
    COORD Cur= {0, 0};
    unsigned long dwLen = 0;

    FillConsoleOutputCharacter(GetStdHandle(STD_OUTPUT_HANDLE), ' ', 80*25, Cur,
    &dwLen);
}
```

- Function to move the cursor

```
void gotoxy(int x, int y)      // move cursor to (x, y)
{
    COORD Pos = {x - 1, y - 1};

    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), Pos);
}
```

Note! You should include windows.h to use these functions.