ITCS222 Computer Organization and Architecture

Section 1: Descending sort five integers from user input

6488181 Thadeeya Duangkaew

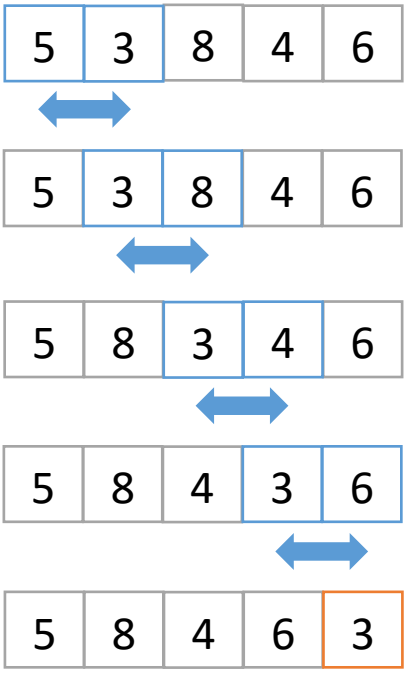
C language - Code

```c
#include <stdio.h>

int main(void) {
  int n = 5;
  int array[n];
  int count;

  printf("Please enter the number of elements in an array: ");
  scanf("%d", &n);

  printf("Please enter the data elements of an array: ");
  for (int i = 0; i < n; i++)
    {
      scanf("%d", &array[i]);
    }

  for (int i = 0; i < n; i++)
    {
      for (int j = i + 1; j < n; j++)
        {
          if (array[i] < array [j])
          {
            count = array[i];
            array[i] = array[j];
            array[j] = count;
          }
        }
    }
  printf("Result Of Descending Order: ");
  for (int i = 0; i < n; i++)
    {
      printf("%d", array[i]);
      printf(" ");
    }
  return 0;
}
```

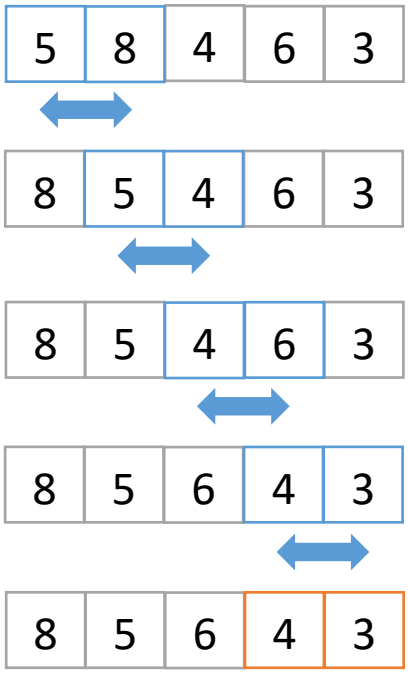
C language – Output

```
> make -s
> ./main
Please enter the number of elements in an array: 5
Please enter the data elements of an array: 5 3 8 4 6
Result Of Descending Order: 8 6 5 4 3 >
```

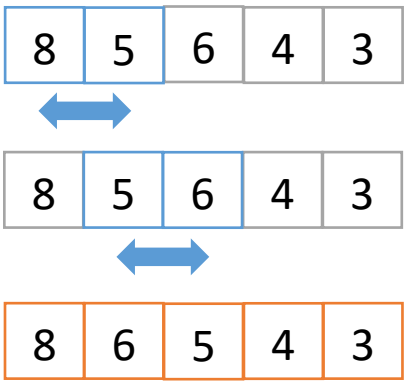
Iteration-1



Iteration-2



Iteration-3



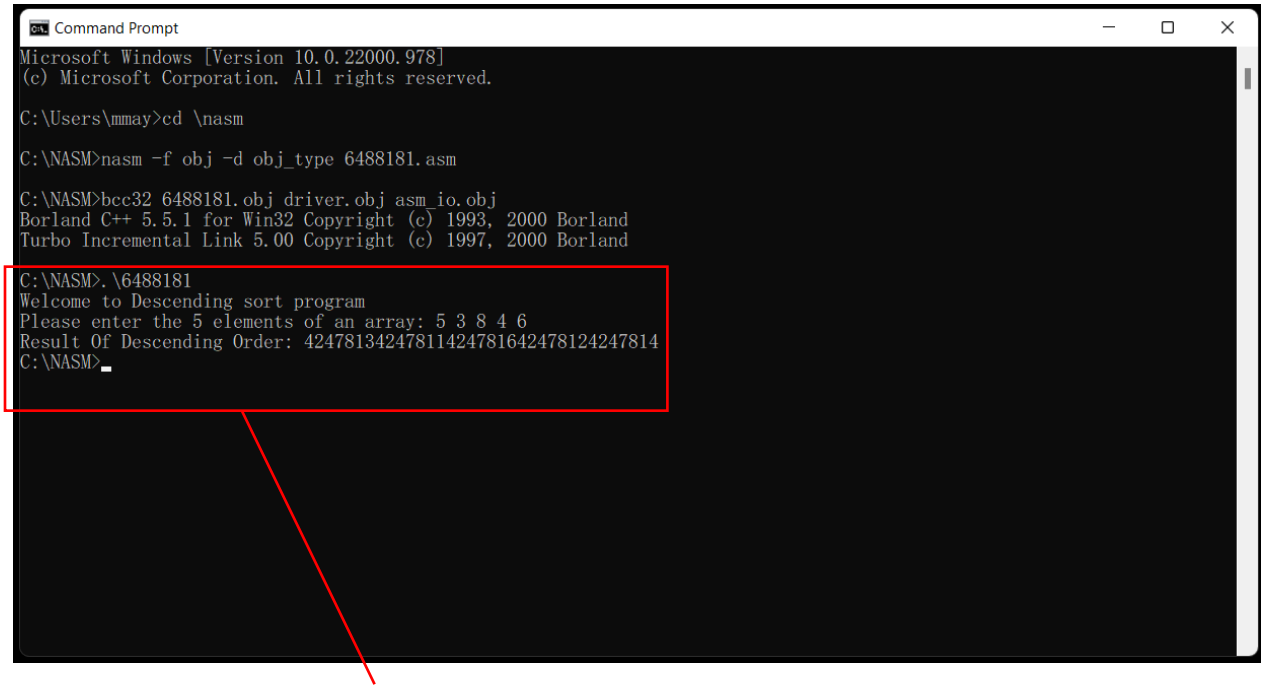
In my thought process I thought I'd like to sort descending order using processes like "Bubble Sort" as a key idea that works by repeatedly swapping the adjacent elements if they are in the wrong order. As shown above.

Assembly language - Code

```
;6488181 Thadeeya Duangkaew
;Section 1: Descending sort five integers from user input

%include "asm_io.inc"

segment _DATA public align=4 class=DATA use32
      msg1      db      "Welcome to Descending sort program", 0
      msg2      db      "Please enter the 5 elements of an array: ", 0
      msg3      db      "Result Of Descending Order: ", 0
      array times 5 db 0

segment _BSS public align=4 class=BSS use32

group DGROUP _BSS _DATA

segment _TEXT public align=1 class=CODE use32
      global  _asm_main
_asm_main:

      mov       eax,  msg1
      call      print_string            ; print message 1
      call      print_nl                ; print new line
      mov       eax,  msg2
      call      print_string            ; print message 2
      mov       esi, 0                  ; count

loop_input:
      call      read_int                ; scanf("%d", & array[i])
      mov       [array+esi], eax        ; when you read int will assign to eax then move to arr[count]
      inc       esi                     ; count++
      cmp       esi,5                   ; input 5 number
      jl        loop_input              ; if less than 5 will go loop
      mov       esi, 0                  ; other count

      mov       eax,  msg3
      call      print_string            ; print message 3

loop_output:
      mov       al, [array+esi]
      call      print_int
      inc       esi
      cmp       esi, 5
      jl        loop_output

      ret
```

Here's the Assembly code that I've try to convert from C language.

Assembly language - Output



The output does not match the C language.

I ran the program in the command prompt, but the assembly code I wrote had some errors so it couldn't run as expected like in C code. I've commented behind the assembly code to make the code easier to write on the next lines.

Source used to write this project

[1] H. Casanova. [Online]. Available: https://courses.ics.hawaii.edu/ReviewICS312/morea/FirstProgram/ics312_nasm_first_program.pdf.

[2] U. o. V. C. Science, 8 March 2022. [Online]. Available: https://www.cs.virginia.edu/~evans/cs216/guides/x86.html.

[3] "tutorialspoint," [Online]. Available: https://www.tutorialspoint.com/assembly_programming/assembly_arrays.htm.

[4] "tutorialspoint," [Online]. Available: https://www.tutorialspoint.com/assembly_programming/assembly_loops.htm.

[5] "stack overflow," [Online]. Available: https://stackoverflow.com/questions/29545696/declaring-arrays-in-x86-assembly.