# Assembly Programming HW5

Assembly Programming (CSE3030)
(Spring 2019)
10th June, 2019

Instructor: Prof. Youngjae Kim

In this homework assignment, students have to program three tasks. Please carefully read each task and program in assembly language.

### Task I

Make a program that sorts number pairs (X, Y). The program should sort all pairs based on X values and if two pairs' X values are same, then sort it based on Y values. For sorting, you should differentiate the order on odd index pairs and even index pairs. The odd index pairs should be sorted in ascending order for X values and in descending order for Y values in case of same X values between pairs. The even index pairs should be sorted in descending order for X values and in ascending order for Y values in case of same X values.

Assume that pairs are given like (0x51, 0x13), (0x1, 0x1), (0x2, 0x3), (0x1, 0x5), (0x51, 0x21), (0x2, 0x1), (0x1, 0x2).

1st index : (0x51, 0x13)
2nd index : (0x1, 0x1)
3rd index : (0x2, 0x3)
4th index : (0x1, 0x5)
5th index : (0x51, 0x21)
6th index : (0x2, 0x1)
7th index : (0x1, 0x2)

The odd index pairs are (0x51, 0x13), (0x2, 0x3), (0x51, 0x21), (0x1, 0x2). They will be sorted to (0x1, 0x2), (0x2, 0x3), (0x51, 0x21), (0x51, 0x13). The even index pairs are (0x1, 0x1), (0x1, 0x5), (0x2, 0x1). They will be sorted to (0x2, 0x1), (0x1, 0x1), (0x1, 0x5). Therefore, the pairs will be sorted like below at last.

(0x1, 0x2)
(0x2, 0x1)
(0x2, 0x3)
(0x1, 0x1)
(0x51, 0x21)
(0x1, 0x5)
(0x51, 0x13)

Make a file "hw5_1.inc" and declare following data in hw5_1.inc. (The values of data will be changed when TAs test your code)

LenData DWORD 7
ArrData DWORD 510013h, 10001h, 20003h, 10005h, 510021h, 20001h, 10002h

LenData is the number of pairs and ArrData is the given pairs. All pairs are given as double word data type, and their MSB 16 bits are X values and LSB 16 bits are Y values in pair (X, Y). In above, for example, the data 510013h's X value is 51h, and Y value is 13h (00510013h). In your program, you should print all pairs in the given sequence. After then, you should also print all pairs in sorted sequence like below.

```
C:₩>s181234↲
Before sort : 00510013, 00010001, 00020003, 00010005, 00510021, 00020001, 00010002
After sort : 00010002, 00020001, 00020003, 00010001, 00510021, 00010005, 00510013
Bye!
C:₩>
```

Note:

1. All X and Y of pairs are given in 16bit unsigned hexadecimal integer. That is, X and Y are given together in one double word data type variable.
2. All X are larger than 0 (X > 0).
3. Sort data about X values prior to Y values.
4. Print pairs with WriteHex function in Irvine library.
5. Follow the printing format. (e.g. strings to print, comma, etc.)

**Task II**

Make a program that concatenates a source string to the end of a target string. Sufficient space must exist in the target string to accommodate the new characters. Pass pointers to the source and target strings.

```
targetStr BYTE “ABCDE”, 10 DUP(0)
sourceStr BYTE “FGH”, 0
```

NOTE:

1. The length of “target” string and “source” string does not exceed 40 characters.
2. ”source” string and “target” string should be NULL terminated.
3. If the number of characters in “source” string exceeds the number of characters in “target” string, do not print any output.

Ex. target : “ABCED”, source : “FGH”

```
C:₩>s181234↲
Concatenated string : ABCDEFGH
C:₩>
```

### Task III

Make a program that searches for the second matching occurrence of a source string inside a target string and returns the second matching position. If a match is found, print the index of the second matching position of the target string. Otherwise, do not print any output. Make a file "hw5_3.inc" and declare only following data in hw5_3.inc (The values of data will be changed when TAs test your code).

```
target BYTE "123ABC342432ABCD123", 0
source BYTE "ABC", 0
pos    DWORD ?
```

NOTE:

4. The length of "target" string and "source" string does not exceed 40 characters.
5. "source" string and "target" string should be NULL terminated.
6. If the number of characters in "source" string exceeds the number of characters in "target" string, do not print any output.
7. If a match is found, save first index of matching position to "pos" variable and print it.
8. If there are multiple matches, just consider the first match.
9. If there is no match, do not print any output.

Ex. target : "123ABC342432ABCD123", source : "ABC"

```
C:₩>s181234↲
Position value : 12
C:₩>
```

## Submission:

- Submission Due Date: 6/23 (SUN) 11:59 PM (Late : -10% per day, up to 2 days)
- You need to submit compressed **three code files**, each corresponds to each task.
- Name the .asm files with the last 6-digits of your student id and an underscore ('_') and the number of the task. (e.g. **181234_1.asm, 181234_2.asm, 181234_3.asm**)
- Compress source code files into **zip format** and name it with your last 6-digits. (e.g. 181234.zip)
- Please submit your assignment under **Assignment5** in Assignment menu in Cyber Campus.

**Grading Policy:**

- **When a code copy is found, the person showing the code and the person who copied the code will get all zero scores.**
- If there is an assemble error, you will get 0 point.
- You will get minus one (-1) point for each wrong file name.
- Your program will be tested with different keys and texts and pairs.
- You can get higher score if the code + data size is shorter.