# BLG 212E - Microprocessor Systems

Homework 2

Sadettin Fidan

December 17, 2024

**Due Date**: 31.12.2024, Tuesday, 12.00pm

## Preamble


## Introduction

In this section you will get familiar with the homework structure.

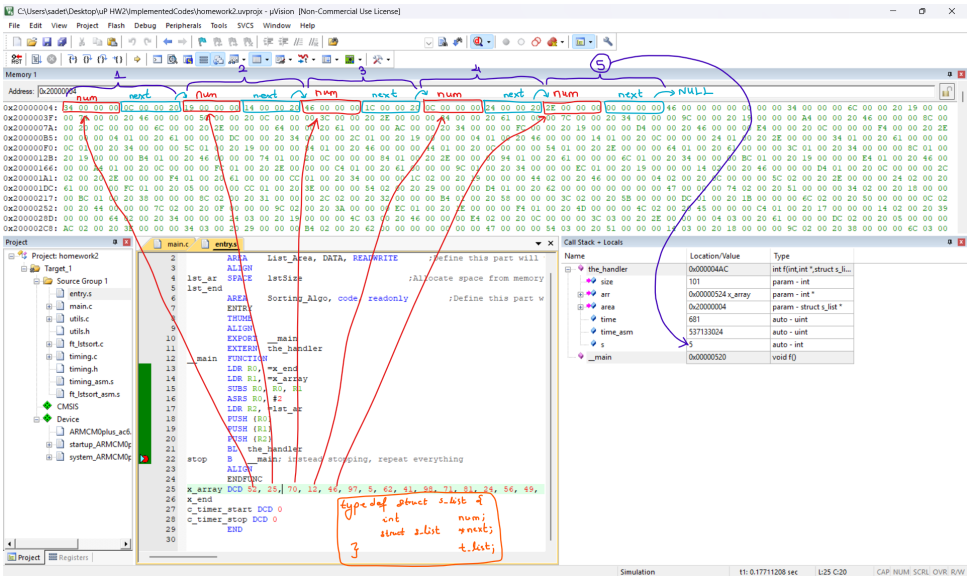

Figure 1: The array and the linked list

### About Figure 1

1. As shown in the Figure 1, an array called x_array is defined with 101 random numbers.

2. As shown in the Figure 1, a region is initialized as starting with the address 0x200000004 to be used for the linked list.

3. As shown in the Figure 1, the region for the linked list is filled by x_array.

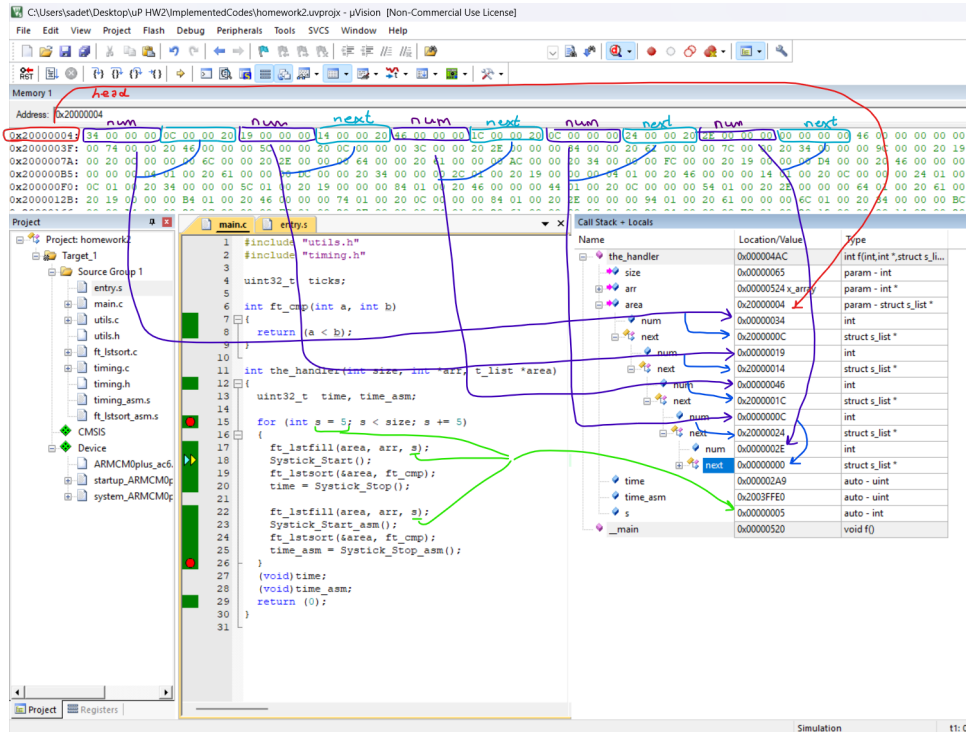


Figure 2: Filling the linked list using the array

### About Figure 2

1. As shown in the Figure 2, ft_lstfill function is used to fill the linked list area using the array.

2. ft_lstfill function takes the size parameter s which is iterating through 5, 10, 15, ..., 100.

3. So, each time ft_lstfill called with s, first s number of integers in the x_array are filled into the linked list to be sorted.

### About Figure 3

1. As shown in the Figure 3, as iterated once in the for loop, one can observe the first 5 integers of x_array is sorted using the linked list.

2. As shown in the Figure 3, time and time_asm are 5. Which means ft_lstsort is executed 50us or thereabouts.

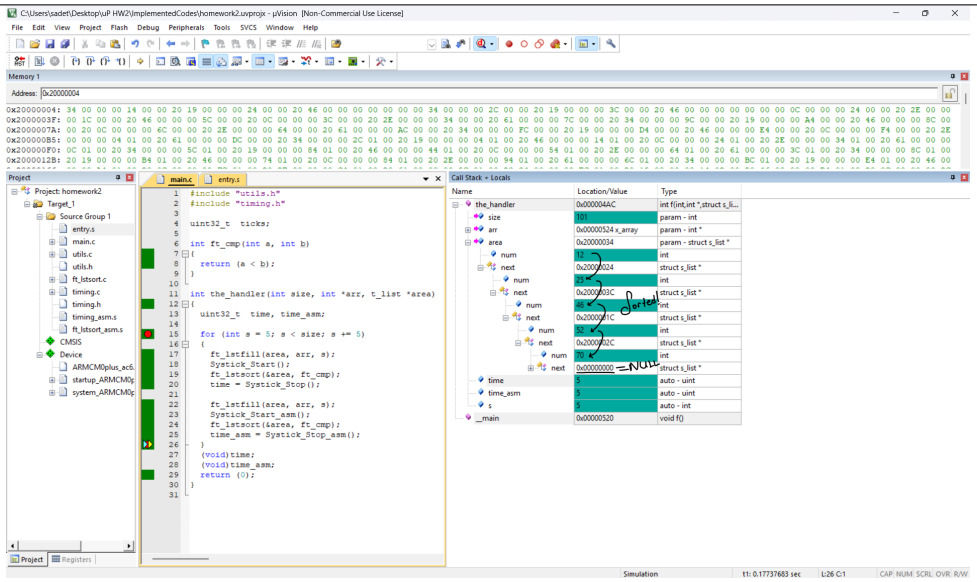

Figure 3: Sorting in the first iteration

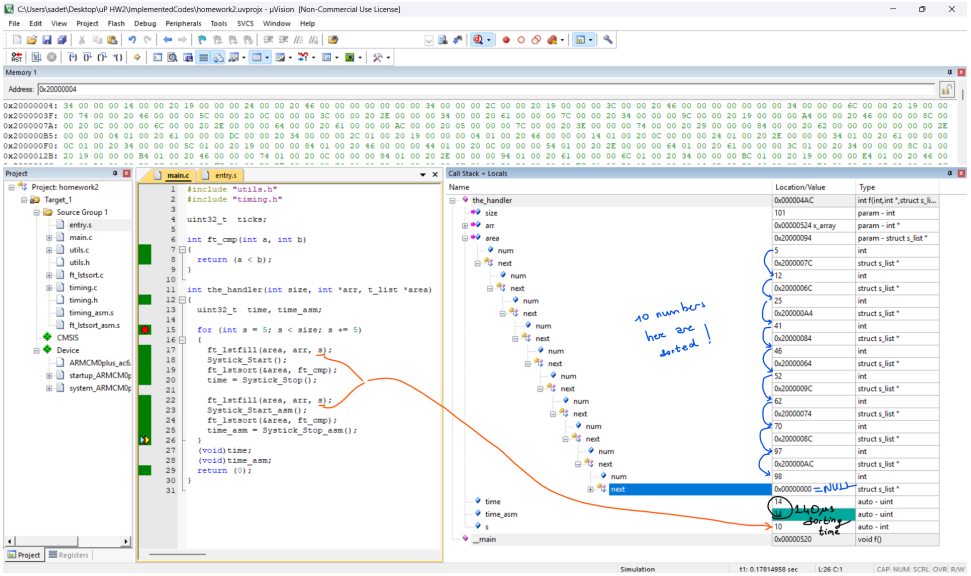

Figure 4: Sorting in the second iteration

3. Since ft_lstsort_asm is not yet implemented and also not yet called, instead ft_lstsort is called, we might obviously expect time and time_asm to be the same in each iteration.

**About Figure 4**

1. As shown in the Figure 4, sorting 10 numbers takes about 140us.

**About Figure 5**

1. As shown in the Figure 5, all 100 numbers are succesfully sorted.

**About Figure 6**

1. In Figure 6, ft_lstlast and ft_lstlast_asm functions are called.

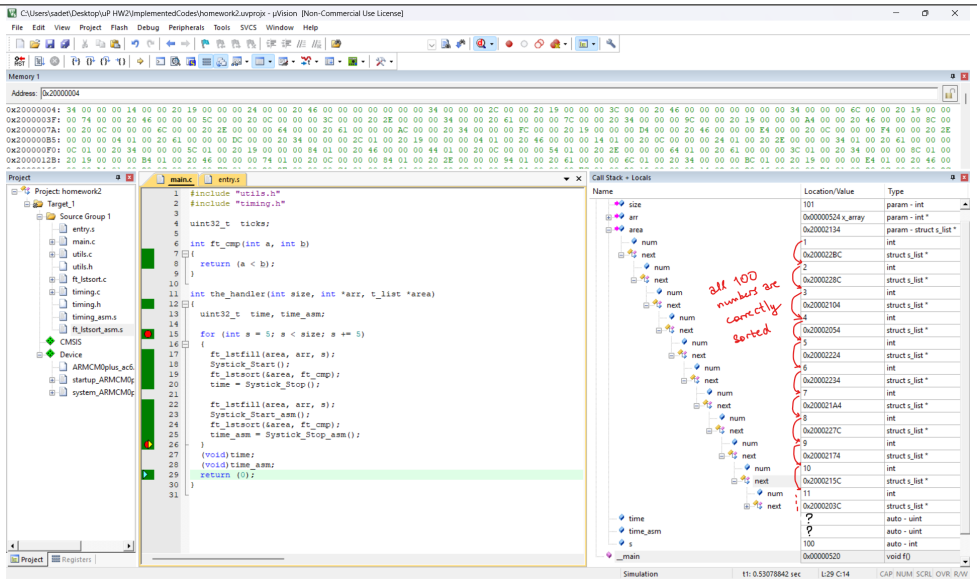


Figure 5: Sorting in the last iteration

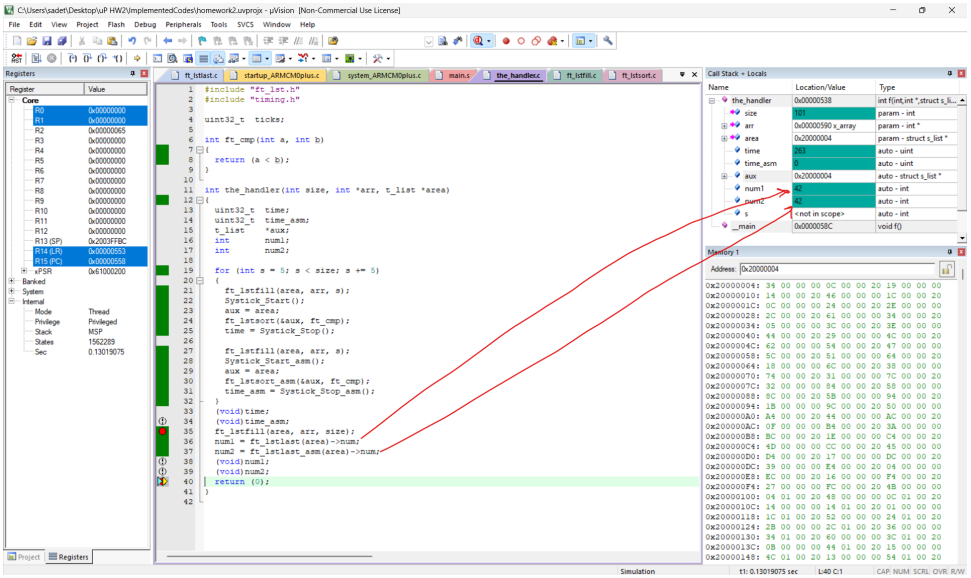

Figure 6: Linked List Last Element

2. These functions are responsible of returning last element of a linked list.

3. ft_lstlast_asm is ARM-CORTEX-M0+ ASM implemented version of ft_lstlast.

**About Figure 7**

1. In Figure 7, ft_lstlast and ft_lstlast_asm implementations are provided.

2. So, one can learn from these implementations and implement ft_lstsort_asm in this homework.

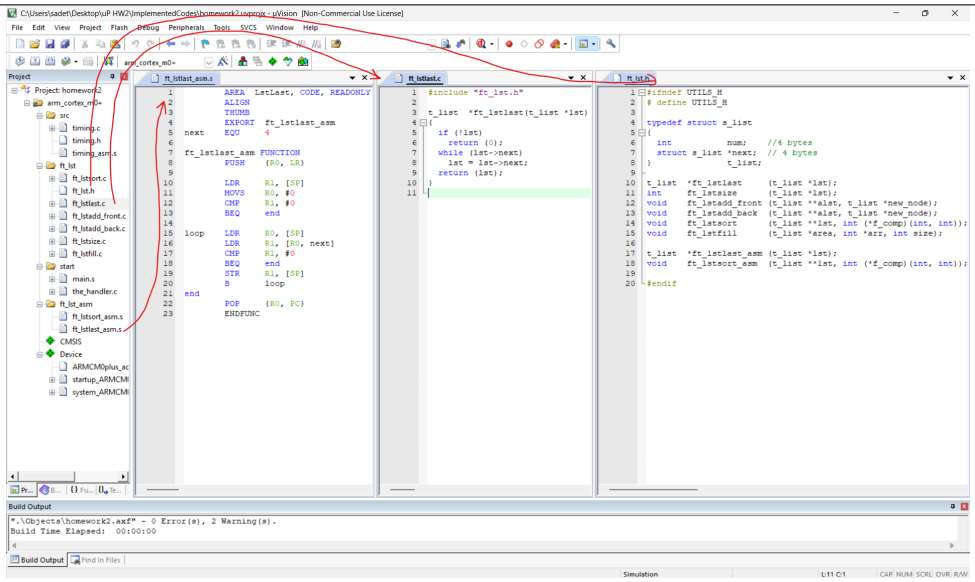

Figure 7: Implementation of ft_lstlast and ft_lstlast_asm

# Requirements

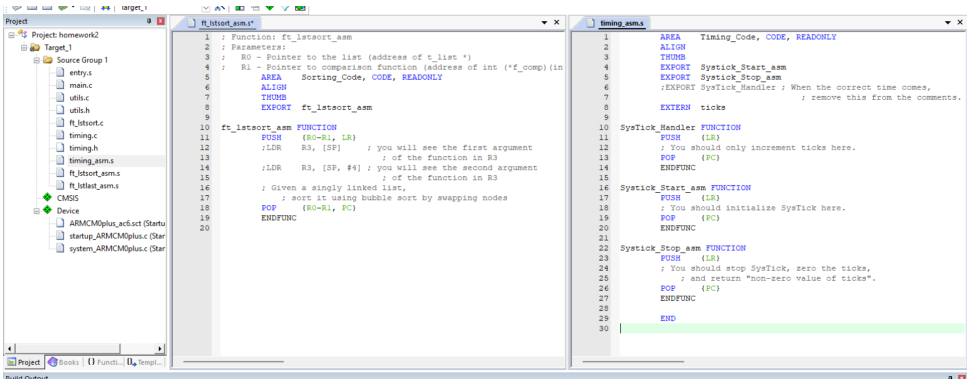

Figure 8: Template codes

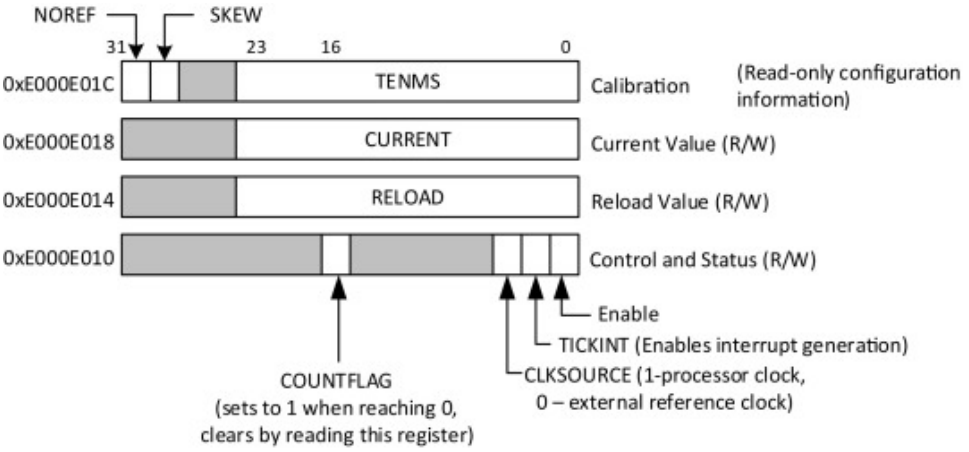


Figure 9: SysTick Registers

1. In Figure 8, templates for timing_asm and ft_lstsort_asm are provided. You should implement these as detailed below:

2. **System Timer (50pts)**

   a You should implement SysTick_Start_asm for the Systick timer to start sending interrupts to the CPU. Check for the registers as shown in Figure 9

   b You should implement SysTich_Stop_asm to stop Systick timer and save ticks value.

   c You should go into timing.c, delete SysTick_Handler's C implementation, come back to timing_asm.s and uncomment ;EXPORT SysTick_Handler, and finally implement Systick_Handler FUNCTION. Thus, you complete timer part.

3. **Sorting Algorithm (40pts)**

   a For sorting side of this homework, you should implement a bubble sort algorithm.

4. **BigO Analysis (10pts)**

   a And finally, if you go back to Figure 4, you will see how to measure time passed for sorting the numbers.

   b You are required to save the measurements for time and time_asm for each iteration, and provide them as a graph in your report.

# Constraints

- Your code should include a comment for each line. Otherwise, points will be deducted.
- If you need to calculate any value, please write your formula and explain it step by step as a comment in the code.
- The program must be implemented with Arm Cortex M0+ assembly language.
- Your assembly source file is expected to work with Keil uVision IDE v5.
- Default configuration must be sufficient to run your programs. If your program expects any different configuration parameter, please write this at the top of the code in comment lines.
- If your program does not run with Keil uVision IDE you will get zero point from this question.

# Submission

- Please submit a single zip file.
- Please put your report inside the zip file as StudentNo_NameSurname_uP_HW2.pdf

- Please put your Keil MDK project under the folder codes/
  - This folder should be the root directory of the project.
  - You should clean all built files.
- You are expected to submit your homework 2 through the Ninova system before the due date. Late submissions will not be accepted unless submission date change.

Any solution must be your own work. If any plagiarism is detected, disciplinary regulations of the university will be followed.