EE 337: Microprocessors Laboratory (Spring 2022)
Indian Institute of Technology Bombay

Assignment 3: 20 points Date: January 25, 2022

1. [20 points] In this lab, you will write an assembly language program to perform binary search (`https://www.geeksforgeeks.org/binary-search/`). There will be an array in memory consisting of a **strictly increasing sequence** of 8-bit unsigned integers. This array has to be searched for the presence of a given 8-bit query integer. If the query integer is present in the array, the program should return the memory location of the integer. To add the array to memory, you will be using the Keil command window as described below.

   Use the following program as a starting point. Add your code in the `SEARCH` subroutine.

   ```
   ORG 0H
   LJMP MAIN

   ORG 100H
   MAIN:
   CALL SEARCH

   HERE: SJMP HERE

   ORG 130H
   SEARCH:
   // Add your code here.
   RET
   END
   ```

   - The starting address of the array `ArrayAddress` will be stored in location `30H`. The starting address is only one byte long which can take values from `80H` to `FFH`.
   - You can assume that the array consists of strictly increasing values.
   - The length `N` of the array will be stored in location `31H`.
   - The element `S` being searched will be stored in location `32H`.
   - If the element being searched is present in the array, its location needs to be stored in location `33H`.
   - If the element being searched is not present in the array, you should store `0EH` at the location `33H`. As the array occupies locations in the range `80H` to `FFH`, the location `0EH` will never be returned if the element is present in the array.
   - The array length `N` can be zero. You should handle this case by returning `0EH` in the location `33H`.
   - To reduce the effort involved in adding multiple items in memory locations, we can use the command window in Keil.
     - Start a Keil debugging session.
     - Modify the memory locations `30H`, `31H`, and `32H` to store `ArrayAddress`, `N`, `S` respectively.
     - For `ArrayAddress` = `90H`, enter the following command in the Keil command window to load an array of 16 numbers represented in decimal format.

The I:90h refers to indirect addressing of location 90H. To inspect the memory, you should enter I:0x90 in the Keil memory window.

```
E char I:90h = 7,13,17,19,41,48,90,94,124,141,144,196,202,229,235,249
```

The following test cases show expected results.

- For ArrayAddress = 90H, N = 10H, S = 5EH, the answer in location 33H should be 97H. This is because the number being searched (94 in this case) is in location 97H.
- For ArrayAddress = 90H, N = 10H, S = 80H, the answer in location 33H should be 0EH. This indicates that the element being searched (128 in this case) is not present in the 16 locations starting from location 90H.

- Here is a longer array.

```
E char I:80h = 23,30,33,45,54,58,62,68,70,81,82,86,87,88,91,93,
94,102,107,111,132,137,139,157,161,169,176,177,179,190,193,194,
201,206,215,230,232,247,248,255
```

The following test cases show expected results.

- For ArrayAddress = 80H, N = 28H, S = FFH, the answer in location 33H should be A7H.
- For ArrayAddress = 80H, N = 23H, S = FFH, the answer in location 33H should be 0EH. This indicates that the element being searched (255 in this case) is not present in the 35 locations starting from location 80H.

TA Checkpoints

1. Check the four test cases shown above.
2. Check that the program handles the case of N=0.
3. Ask the student to explain the code in the SEARCH subroutine.