# Project 3

CS 17

Professor Haghoo

1

For Faculty Use Only: Colors: Office 2007-2010

# Project 3



Write a program to read some numbers.

- **The program saves the number into heap memory.**
- The program stops when a zero is entered.
- **Program displays the numbers in the order of input in a single line.**
- Make sure that you print top and bottom lines with your own name with appropriate new lines. (Also see the next slide)

```
Student Name - Project 3

Enter a positive number, 0 to stop: 45
Enter a positive number, 0 to stop: 67
Enter a positive number, 0 to stop: 23
Enter a positive number, 0 to stop: 3
Enter a positive number, 0 to stop: 90
Enter a positive number, 0 to stop: 56
Enter a positive number, 0 to stop: 87
Enter a positive number, 0 to stop: 51
Enter a positive number, 0 to stop: 44
Enter a positive number, 0 to stop: 49
Enter a positive number, 0 to stop: 65
Enter a positive number, 0 to stop: 98
Enter a positive number, 0 to stop: 79
Enter a positive number, 0 to stop: 0

45 67 23 3 90 56 87 51 44 49 65 98 79

Student Name - End of Project 3
```

# Project 3 –Details

- First print your name, and the project number. See the previous window.
- Then prompts and read the numbers.
- Work with 32-bit unsigned integers.
- No error checking.
- Test the program with data which are within the limits.
- Follow the previous programs style and commenting.

Stay on the same line, i.e., keep prompt and the answer on one line.

# Project 3 – Notes

- Use the **heap memory** (not stack) for the data.
- You may use stack for some individual instructions.
- After allocating memory, make sure that that:
  - You save EAX, so you can retrieve its original value whenever is needed.
  - EAX will be used by HLA for many other things.
  - Make sure that you release the allocated memory before ending the program.
  - It is recommended that you copy the following lines right at the beginning. You may test your program many times, so you release the memory each time.

```
mem.alloc (1000);                    // Allocate memory
mov (EAX, memPointer);               // Save EAX



mov (memPointer, EAX);               // Retrieve EAX's original value
mem.free (EAX);                      // Release memory
```

This hint is a suggestion. You don't have to follow.

# Project 3 – Hints



Read Chapter 3 very carefully.

The program should have two independent loops. (There are other solutions possible.)

- First one to read numbers and save them to the heap memory.
- Second one to print them from the heap memory.
- For your information, my own program is about 55 lines out of which only 30 lines are real code.

- This is simple program but contains delicate instructions, likely you are using them for the first time. Don't delay. Begin sooner.
- I will try to answer and help you, provided that you ask your questions soon enough.
- I can not answer questions such as: I do everything like the book but my program is not working. Practically, nobody can help it. Questions must be specific.
- Develop the program gradually. Save each working version on a different file, so you can always have something done. Submit the most completed one.

# Project 3 – Last Slide