



SUMMER SESSION I 2014
EECS 10 WEEK4 DISCUSSION1
Che-Wei Chang

OUTLINE

- Assignment 4 Part1
 - Option 6-8(9) of Menu Driven Calculator for floating point number [50pts]
- Only **ONE** set of program, script, and txt file for this assignment.



ASSIGNMENT 4

- Calculator
 - Deadline : 07/21/2014
- Name your files `calculator.c`, `calculator.txt`, and `calculator.script`
- Make sure your program is free of warning
 - Use `-Wall` option to show all warnings.
- Hints
 - Before you implement your work, review lecture slides about `function declaration`, `function definition`, and `function call`.
 - Read the assignment handout carefully
- Menu driven calculator for floating point number
 - Expend the functionality of the current calculator
 - SquareRoot(20pts), NthRoot, Pi(20pts), Cosine(10pts)
 - Use `cos()` function from `math.h`



MENU DRIVEN CALCULATOR

- Prompt a menu and user can choose the operation
 - Operation List
 - 1. Add
 - 2. Subtract
 - 3. Multiply
 - 4. Divide
 - 5. Absolute
 - 6. Square Root
 - 7. Pi
 - 8. cosine
 - 9. N-th root (Bonus)
 - 10. Quit
- Part 1
- Part 2



MENU DRIVEN CALCULATOR

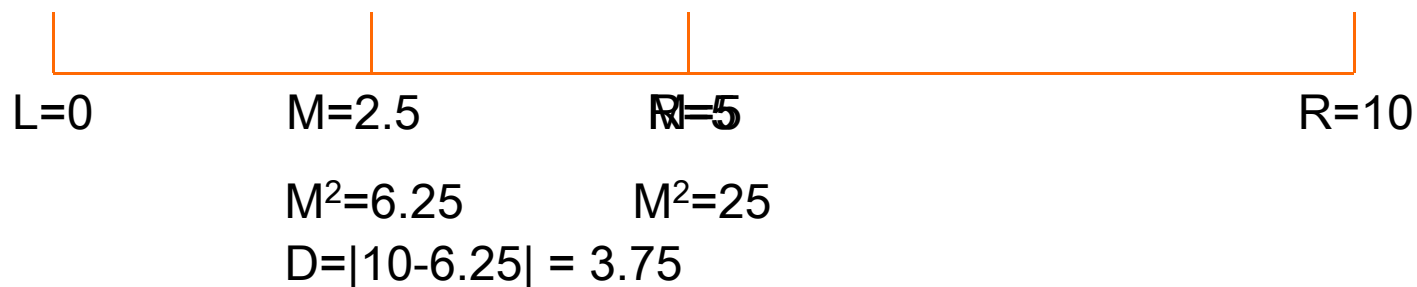
- Should be aware of the following points:
 - The execution only finishes when the user want to quit
 - Infinite loop, with breaking condition
 - Asking for additional operand if required
 - Ex: user chooses square root → 2nd operand is not needed.
 - Ex: user chooses N-th root → asking for the 2nd operand.
 - A function has to be created for the corresponding operation
 - Ex:

```
/*function ApproximateTan*/  
double ApproximateSquareRoot(double x) ;  
double ApproximateNthRoot(double x, int n) ;
```



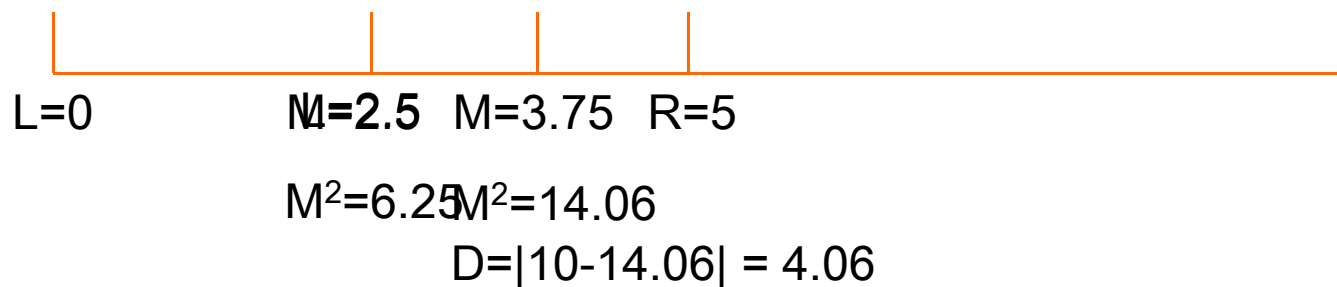
SQUARE ROOT CALCULATION

- Assumption
 - input is a positive natural number (ex. 0, 1, 2, 3, 4, ...)
- Input / Output
 - Input: positive natural number
 - Output: square root of the input number
- Algorithm
 - Binary search approximation
 - Ex. Square root of 10



SQUARE ROOT CALCULATION

- Assumption
 - input is a positive natural number (ex. 0, 1, 2, 3, 4, ...)
- Input / Output
 - Input: positive natural number
 - Output: square root of the input number
- Algorithm
 - Binary search approximation
 - Ex. Square root of 10



SQUARE ROOT CALCULATION

- Assumption
 - input is a positive natural number (ex. 0, 1, 2, 3, 4, ...)
- Input / Output
 - Input: positive natural number
 - Output: square root of the input number
- Algorithm
 - Binary search approximation
 - Ex. Square root of 10



$$L=2.5 \quad \cancel{M=3.75} \quad R=5$$

$$M=3.125$$

$$M^2=9.765$$

$$M^2=9.765$$

$$D=|10-9.765| = 0.235$$



SQUARE ROOT CALCULATION

- Assumption
 - input is a positive natural number (ex. 0, 1, 2, 3, 4, ...)
- Input / Output
 - Input: positive natural number
 - Output: square root of the input number
- Algorithm
 - Binary search approximation
 - Ex. Square root of 10
 - Terminate the binary search when
 - $D < 0.00001$



SQUARE ROOT CALCULATION

○ Pseudo Code

- Start with a range of 0 to N
- As long as the range is not accurate enough, repeat the following steps:
 - Compute the middle of the range
 - Compare the square of the middle value with N
 - If the middle value is less than the square root
 - Use middle-to-right as the new range
 - Otherwise
 - Use left-to-middle as the new range
- Output the middle of the latest range as result

- What if we want to compute N-th Root ?



ASSIGNMENT DISCUSSION

- Bonus – N-th root calculation
 - Prompt a message and ask user the value of N
- Briefly describe the control flow for your menu-driven calculator program in txt file
- Name your files `calculator.c`, `calculator.txt` and `calculator.script`.

