

## Inputs

- 10 array Integers from Keyboard
- 1 clamp value integer from keyboard

## Outputs

- Message asking for user to input 10 array values
- Message displaying array
- Message stating calculating average
- Message stating dividing occurring
- Message displaying divided array
- Message stating clamping occurring
- Message displaying clamped array
- Message asking if user wants to continue program or exit

## Variables

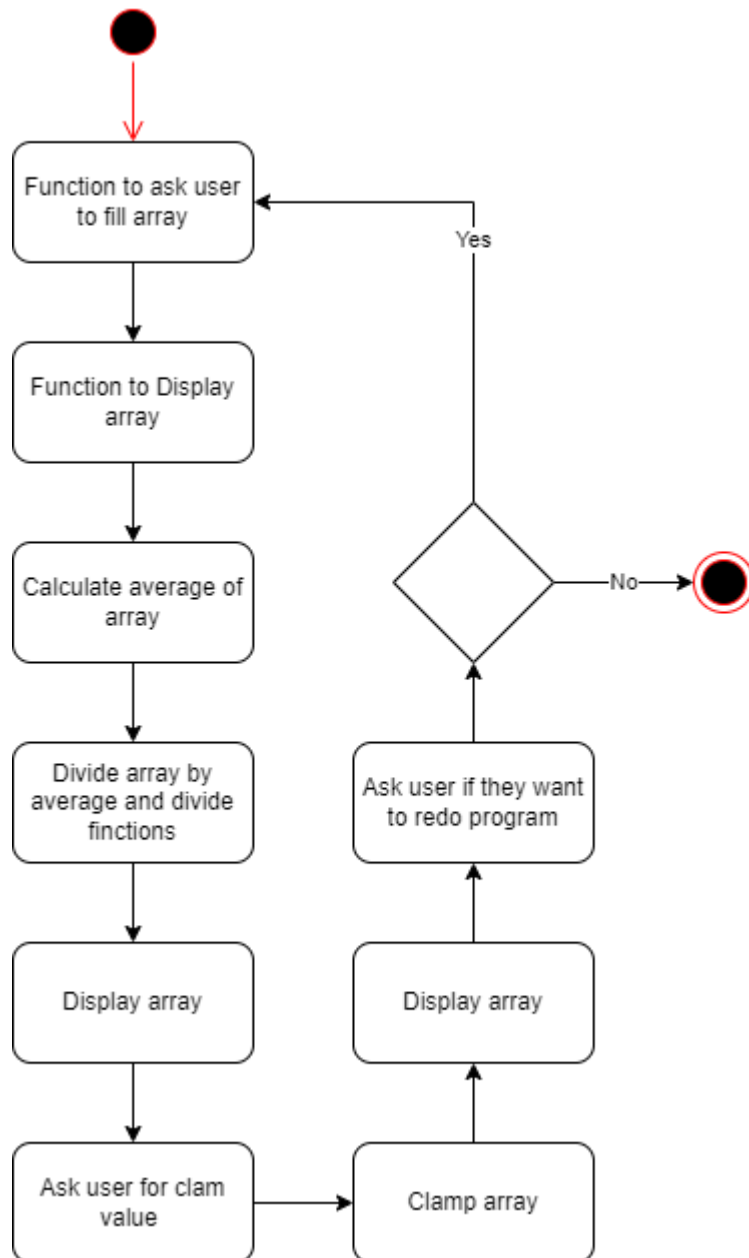
- Input – array of size 10, for integers: arrar
- Output – BYTE for saying display function occuring: msgDisplay
- Output – BYTE for saying display function occuring: msgDisplay
- Output – BYTE message for saying input function occuring: msgInput
- Output – BYTE message asking for array input integer: msgInputValue
- Output – BYTE for saying average function is occuring: msgAverage
- Output – BYTE for saying divide method is occuring: msgDivide
- Output – BYTE message for saying clamping function is occuring: msgClamp
- Output – BYTE message for asking user for clamp integer: msgInputClamp
- Output – BYTE for asking user if they want to exit program: msgExit
- Output – BYTE for formatting, space and comma: msgSpace
- Output – BYTE for formatting, goes to next line: msgLine

## Algorithm

1. Loop 10 times, asking user for 10 integers to fill array with
2. Loop through array and display each value
3. Calculate average of array by summing all array values together, and dividing by 10
4. Divide each integer in array by that averahe
5. Display array
6. Ask user for a clamp value

7. Set the max value of each integer in array to that clamp value
8. Display array
9. Ask user if they want to exit array, or continue (go to step 1)

### Flow Diagram



## Stack Diagrams

### Input Function

Old EBP
EDX
ECX
EBX
EAX
FLAGS <-ESP

### Display Function

Old EBP
EDX
ECX
EBX
EAX
FLAGS <-ESP

### Average Function

Old EBP
EDX
ECX
EBX
FLAGS <-ESP

### Divide Function

Will store ecx
eax
Old EBP
EDX
ECX
EBX
EAX
FLAGS <-ESP

Ariel Sischy  
221003350  
Prac 05

## Clamp Function

Old EBP
EDX
ECX
EBX
EAX
FLAGS <- ESP