3/16/2023 Andrew Davison

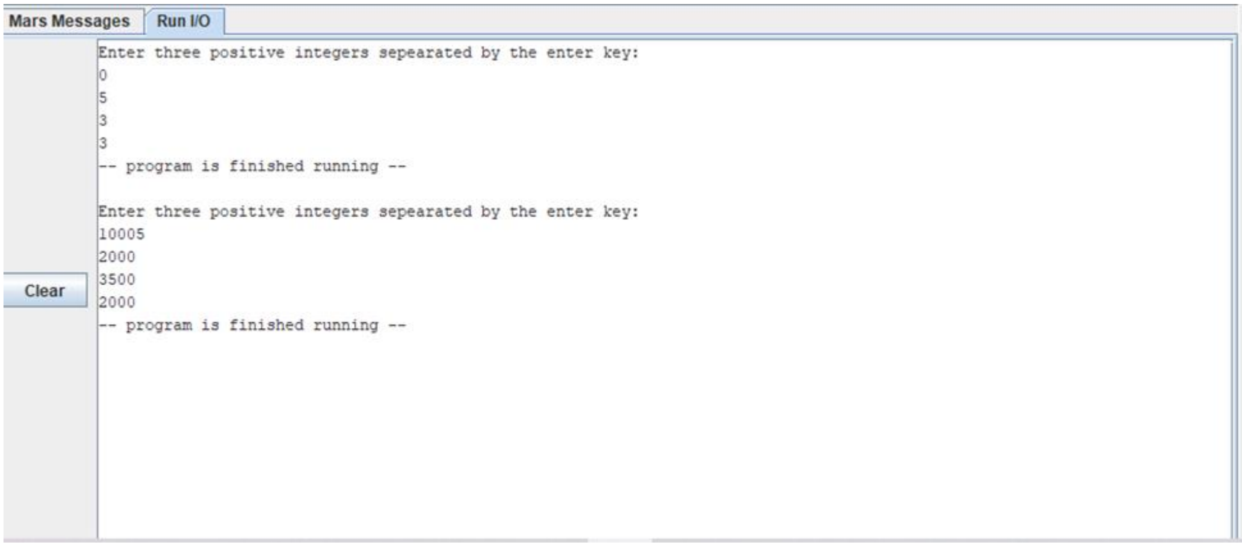
# Project 2

Program 1:

1. The program will have three integers as inputs by the user, and will output the smallest number of the three that is greater than 0
2. The code begins by outputting a greeting to the user to input the three integers. Then begins the branching to find the smallest. The code compares the first two numbers to zero, and then compares them too each other. Depending on which one is smaller, it will branch to a procedure, that compares that number to the third and final number and based on that, outputs which number is the smallest.
3.

| Column1 | Column2 |
|---|---|
| $v0 | return value of syscall |
| $a0 | argument of syscall |
| $t0 | store the first int |
| $t1 | store the second int |
| $t2 | store the third int |
| secNumLastNum | compare the second int and last int |
| firstNumLastNum | compare the first int and last int |
| firstNum | if the first int is the smallest |
| secNum | if the second int is the smallest |
| lastNum | if the last int is the smallest |
| end | shutdown program |

4. Leaf-procedures, User-input, Conditionals, Shutdown program, Jumping procedures
5.

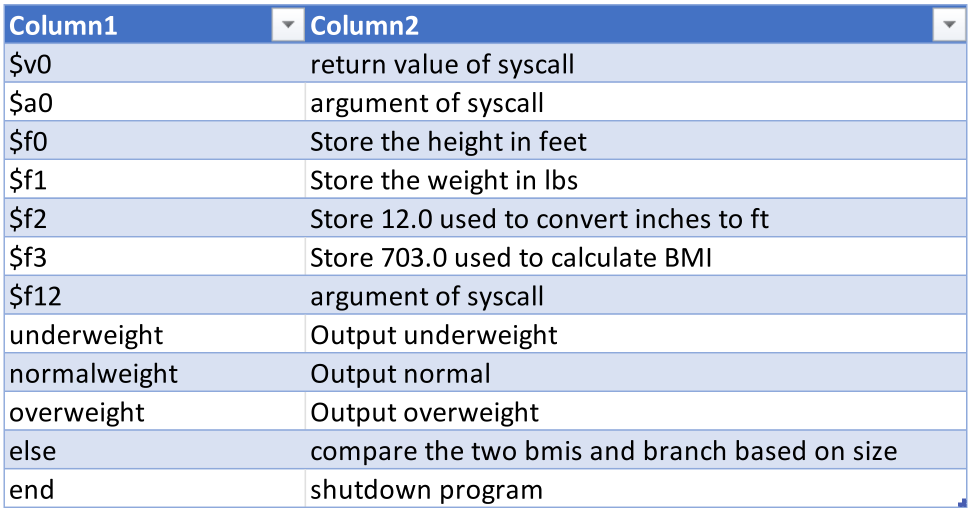


```
Enter three positive integers sepearated by the enter key:
0
5
3
3
-- program is finished running --

Enter three positive integers sepearated by the enter key:
10005
2000
3500
2000
-- program is finished running --
```

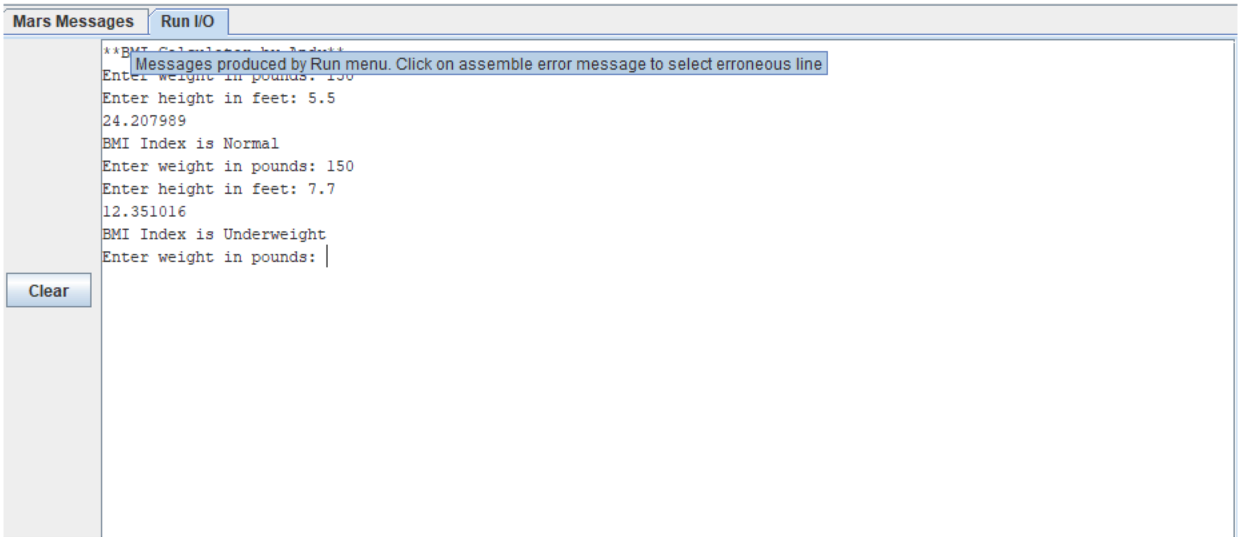
Program 2:

1. The program will ask for 2 different single-floating point inputs from the user. The program will output the calculated BMI for those numbers.
2. The code begins by asking the user to input 2 single-precision floating point numbers. The program then finds the inches from the feet provided by multiplying that by 12. Then it calculates total BMI by squaring the inches, then dividing the weight by that. At the end you multiply it all by 703 to find the BMI. Then the program uses branching procedures to determine if that BMI is underweight, normal, or overweight.
3.



| Column1 | Column2 |
|---|---|
| $v0 | return value of syscall |
| $a0 | argument of syscall |
| $f0 | Store the height in feet |
| $f1 | Store the weight in lbs |
| $f2 | Store 12.0 used to convert inches to ft |
| $f3 | Store 703.0 used to calculate BMI |
| $f12 | argument of syscall |
| underweight | Output underweight |
| normalweight | Output normal |
| overweight | Output overweight |
| else | compare the two bmis and branch based on size |
| end | shutdown program |

4. Leaf-Procedures, Floating point input, Floating point arithmetic, Floating point output, Floating point branching
5.



Messages produced by Run menu. Click on assemble error message to select erroneous line

```
**BMI Calculator by Andy**
Enter weight in pounds: 150
Enter height in feet: 5.5
24.207989
BMI Index is Normal
Enter weight in pounds: 150
Enter height in feet: 7.7
12.351016
BMI Index is Underweight
Enter weight in pounds: 
```

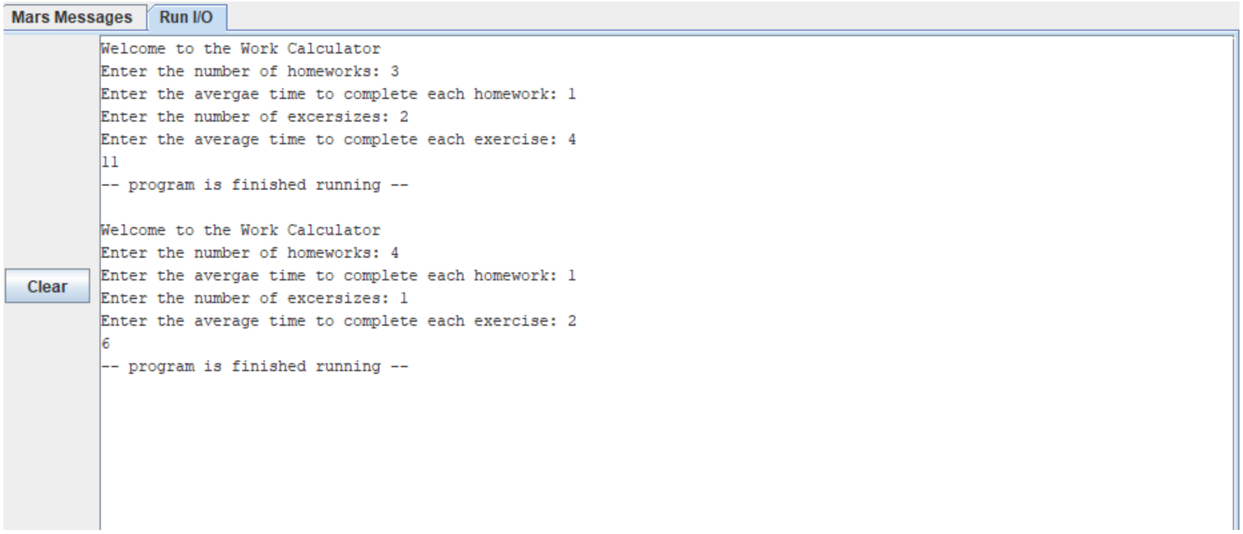
Program 3:

1. The program has 4 integers as input and outputs one integer representing total time

2. The program will ask the user to input 4 integers, representing number of hws and exercises and the avg time to do either. The program then passes those integers into a non-leaf procedure, where it calculate the total time by multiplying then adding the integers.
3.

| Column1 | Column2 |
|---|---|
| $v0 | return value for syscall |
| $a0 | argument of syscall, pass $t0 into procedure |
| $t0 | num of hw |
| $t1 | avg time of hw |
| $t2 | num of exercises |
| $t3 | avg time of exercises |
| $a1 | pass $t1 into procedure |
| $a2 | pass $t2 into procedure |
| $a3 | pass $t3 into procedure |
| $sp | stack pointer to allocate and deallocate memory |
| $ra | return address for procedure |

4. Non-leaf procedure, User input, Allocate space on stack, Deallocate space on stack, Passing registers into procedure
5.



```
Welcome to the Work Calculator
Enter the number of homeworks: 3
Enter the avergae time to complete each homework: 1
Enter the number of excersizes: 2
Enter the average time to complete each exercise: 4
11
-- program is finished running --

Welcome to the Work Calculator
Enter the number of homeworks: 4
Enter the avergae time to complete each homework: 1
Enter the number of excersizes: 1
Enter the average time to complete each exercise: 2
6
-- program is finished running --
```