Cumulative Test: LabVIEW Basics

Test your "Basic" understanding of LabVIEW concepts by attempting the LabVIEW Basics Test. This test is recommended after you complete the seven modules found in the *LabVIEW Basics* section. The test includes 20 multiple choice questions.

1. Numeric controls are different from numeric indicators because they have a ____________ background and ____________.

   a. White, Labels

   b. Grey, Labels

   c. White, Increment/Decrement Buttons

   d. Grey, Increment/Decrement Buttons

2. To switch between the front panel and block diagram, select Window >> Show Block Diagram/Front Panel, or simply press ____________.

   a. Ctrl-E

   b. Ctrl-Z

   c. Ctrl-Shift-Y

   d. Ctrl-B

3. The ____________ function, which allows you to find functions and controls to which you do not know the navigation path, is located on the ____________ palette.

   a. Customize▾, Functions

   b. Search, Controls

   c. Customize▾, Functions and Controls

   d. Search, Functions and Controls

   e. Search, Functions

4. To copy an item on the front panel or block diagram, press <Ctrl-C> and then <Ctrl-V>, or simply hold ____________ and click and drag and drop the item.

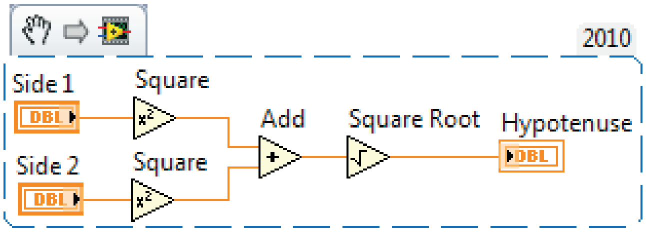
   a. Shift

   b. Altl

   c. Ctrl-Shift

   d. Ctrl

5. What is the correct order of execution of this code?



a. Square Root, Add, Square Functions in Parallel

b. Add, Square Root, Square Functions in Parallel

c. Square Functions in Parallel, Add, Square Root

6. When placing a *new* function, control, indicator, or constant, the _____________ feature wires the terminals together if placed within close enough proximity.

a. Block Diagram Cleanup

b. Automatic Error Handling

c. Automatic Wiring

d. Retain Wire Values

7. The _____________ chooses the best tool based on where the mouse pointer is placed in the LabVIEW environment.

a. Wiring Tool

b. General Tool

c. Automatic Tool Selector

d. Operate Tool

8. The _____________ is typically used during run time to change the value of a control.

a. Operate Tool

b. Automatic Tool Selector

c. Select Tool

d. Shortcut Menu Tool

9. _____________ any object in the LabVIEW environment provides quick access to most of the properties that can be changed.

a. Left-Clicking

b. Right-Clicking

c. Selecting

d. Ctrl-Clicking

10. To find a quick description of an object in the LabVIEW environment, simply turn on _____________ by pressing _____________ and hovering over the object.

a. Detailed Help, Ctrl-I

b. Context Help, Ctrl-U

c. Context Help, Ctrl-H

d. LabVIEW Help, Ctrl-Z

11. A _____________ indicates that the VI is *broken* or not executable.

    [✓ a. Broken Run Arrow]

    b. Run Arrow

    c. Wiring Diagram

    d. VI Icon

12. If you are not getting the expected values from your VI, two good troubleshooting steps are _____________ and _____________, which show wire values during run time and slow down execution, respectively.

    a. List Errors, Highlight Execution

    b. Probing Wires, Using Breakpoints

    [✓ c. Probing Wires, Highlight Execution]

    d. Single stepping, Using Breakpoints

13. Being able to store a list of items in both a string and integer format allows for intuitive programming and a user friendly interface. This is made possible with the _____________ data type.

    [✓ a. Enum]

    b. String

    c. Integer

    d. Cluster

14. The _____________ data structure can be compared to a purse or wallet because a purse or wallet can hold many different things just as this data structure can hold multiple data types.

    a. Array

    [✓ b. Cluster]

    c. Bundle

    d. Container

15. For Loops have auto-index output tunnels, which automatically create _____________ of data at the tunnel.

    [✓ a. Arrays]

    b. Containers

    c. Graphs

    d. Clusters

16. Being able to execute code based on a particular condition allows for logic to be built into a program. In LabVIEW, this is accomplished using a _____________.

    a. While Loop

    b. For Loop

    [✓ c. Case Structure]

    d. Event Structure

17. A _____________ allows a VI to run until a certain condition is met, such as pressing a Stop button on the front panel.

a. While Loop

b. For Loop

c. Case Structure

d. Event Structure

18. A For Loop is different from a While Loop because it runs for a set number of iterations and does not require a _____________. The number of iterations is set by the _____________.

a. Input Tunnel, Count Terminal

b. Conditional Terminal, Count Terminal

c. Conditional Terminal, Iteration Terminal

d. Input Tunnel, Count Terminal

19. To open a prebuilt VI to serve as a starting point or as a point of reference, open _____________ and browse or search for the appropriate VI.

a. LabVIEW Detailed Help

b. NI Example Finder

c. LabVIEW Context Help

d. KnowledgeBase Archive

20. Use _____________ within the LabVIEW environment to see a quick description of any object on the block diagram or front panel.

a. NI Example Finder

b. LabVIEW Manual

c. LabVIEW Context Help

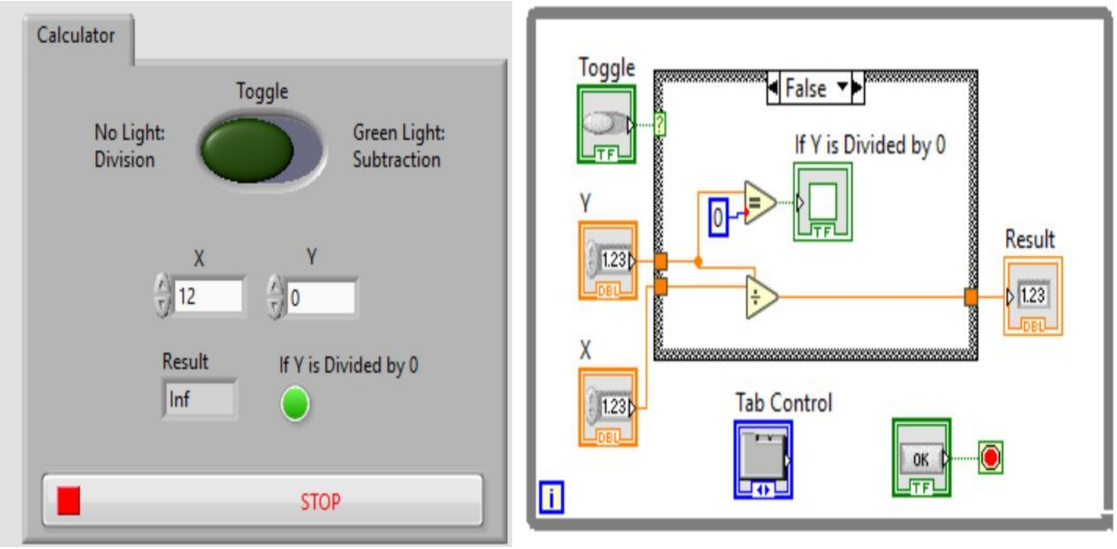
d. LabVIEW Detailed Help

Lab 01 - LabVIEW Short Course

In-Class Problem Set

1. **Division and Subtraction Calculator:**

   The following photo is when it is in ‘Division Mode’ since the On/Off button is off. The Led at the bottom right will light on if Y is divided by 0.

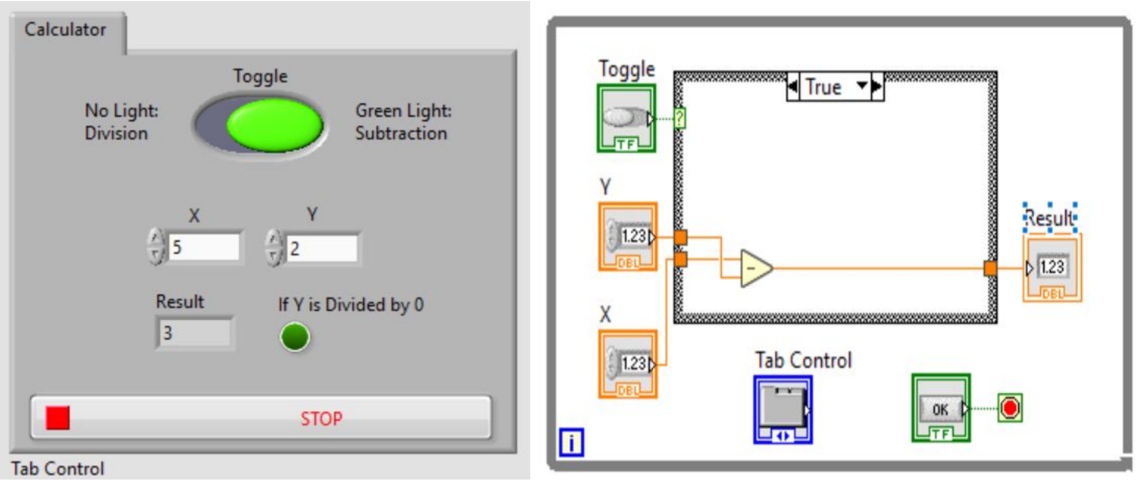


X 12

Y 0

Result Inf

The following photo is when the switch button is turned on indicating that it is in Subtraction mode.

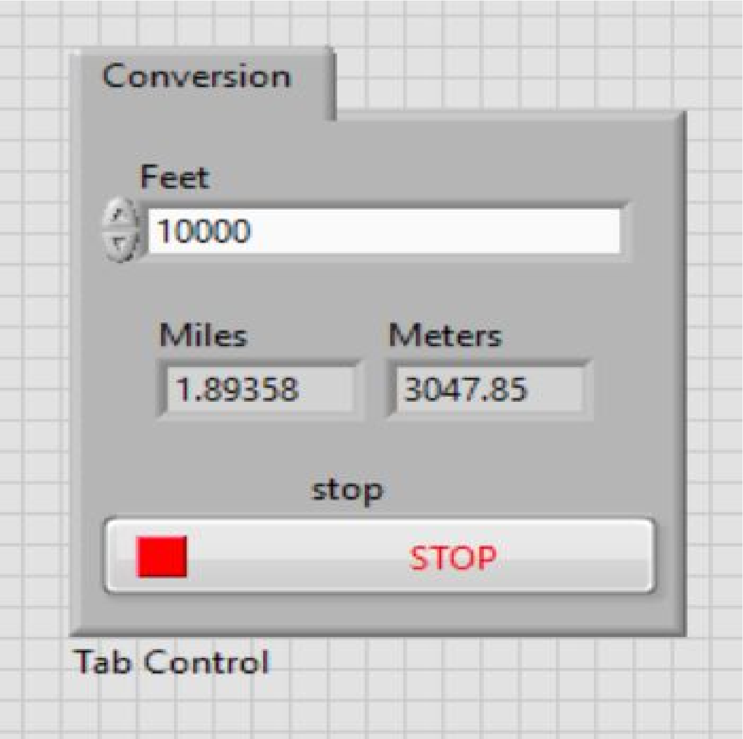


X 5

Y 2

Result 3

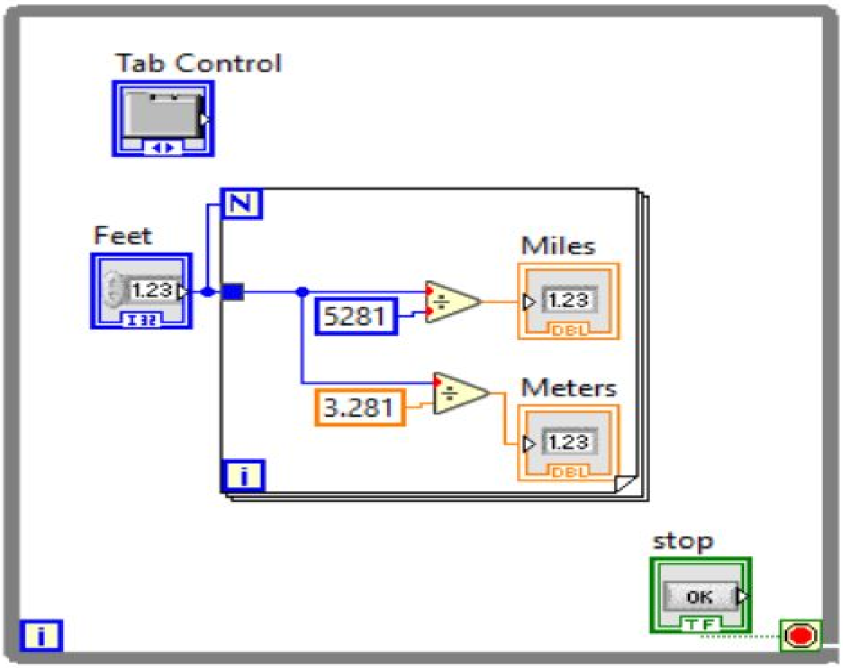
2. **Convert from Feet to Meters and Miles:**



Feet 10000

Miles 1.89358

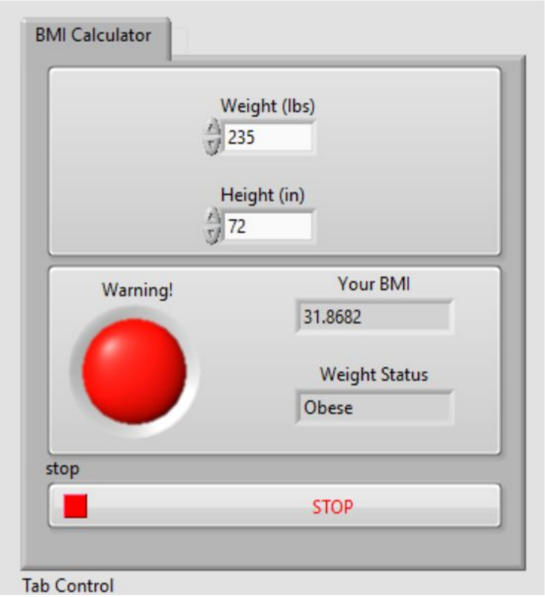
Meters 3047.85



Lab 01 - LabVIEW Tutorials

Take Home Problem Set

1. **Create a VI which will calculate the Body Mass Index (BMI) of an individual.**


BMI Calculator

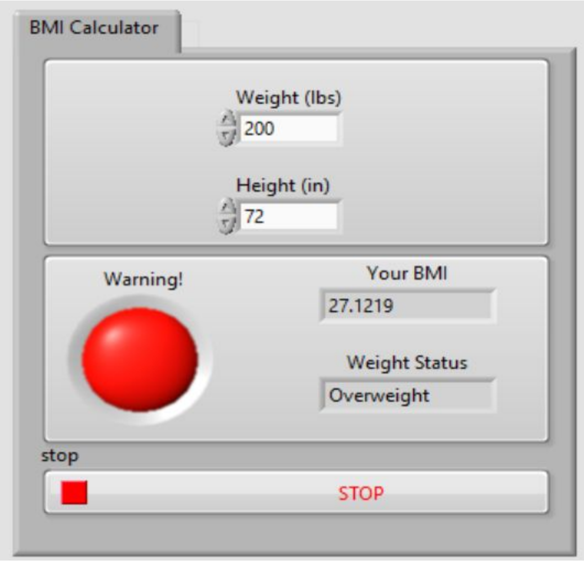
Weight (lbs): 235

Height (in): 72

Warning!

Your BMI: 31.8682

Weight Status: Obese


BMI Calculator

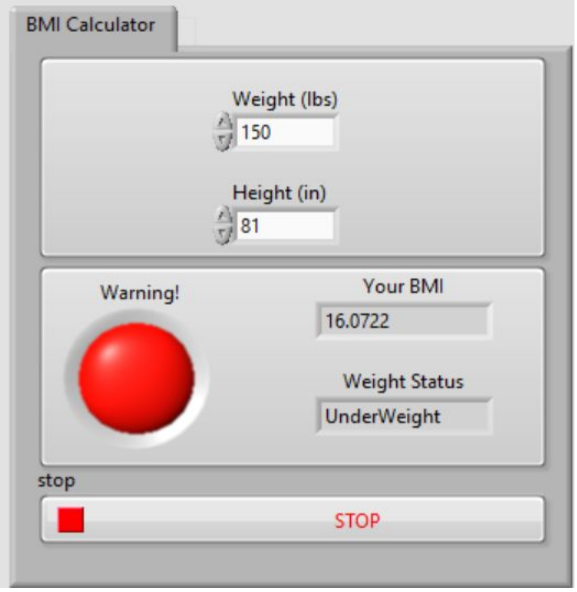
Weight (lbs): 200

Height (in): 72

Warning!

Your BMI: 27.1219

Weight Status: Overweight


BMI Calculator

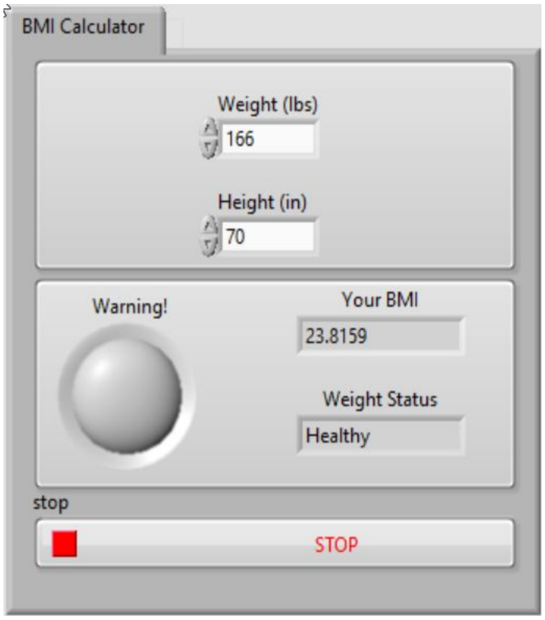
Weight (lbs): 150

Height (in): 81

Warning!

Your BMI: 16.0722

Weight Status: UnderWeight


BMI Calculator

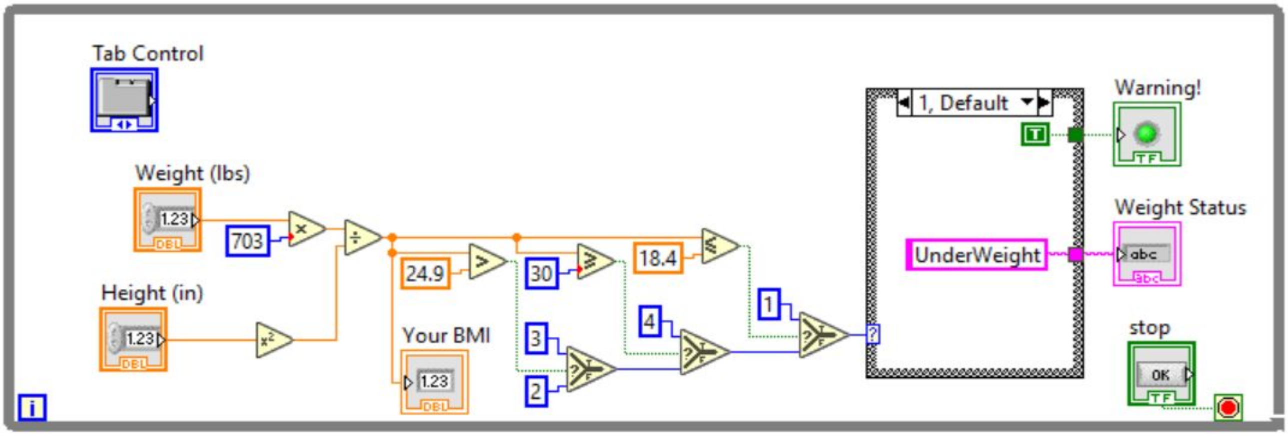
Weight (lbs): 166

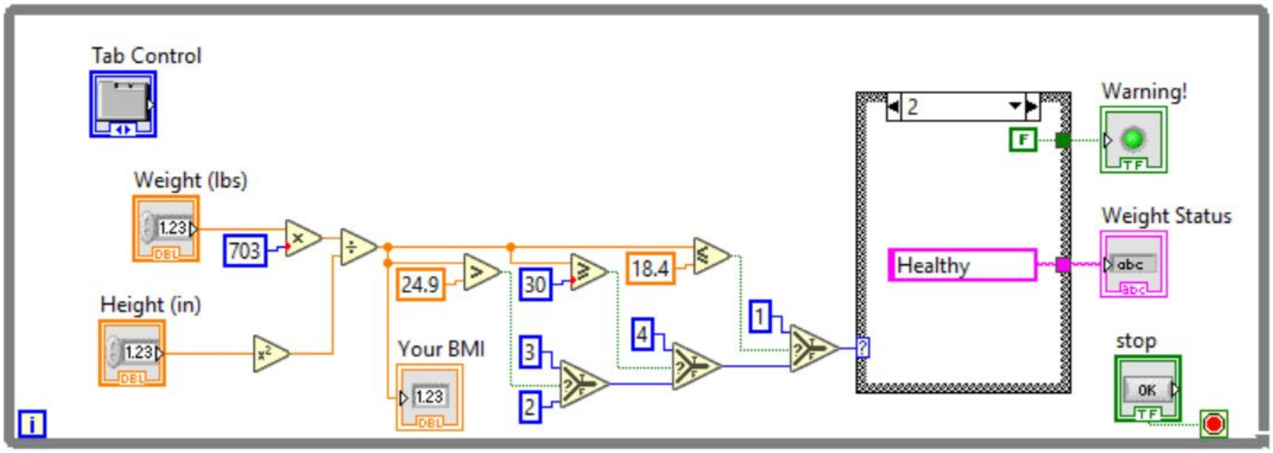
Height (in): 70

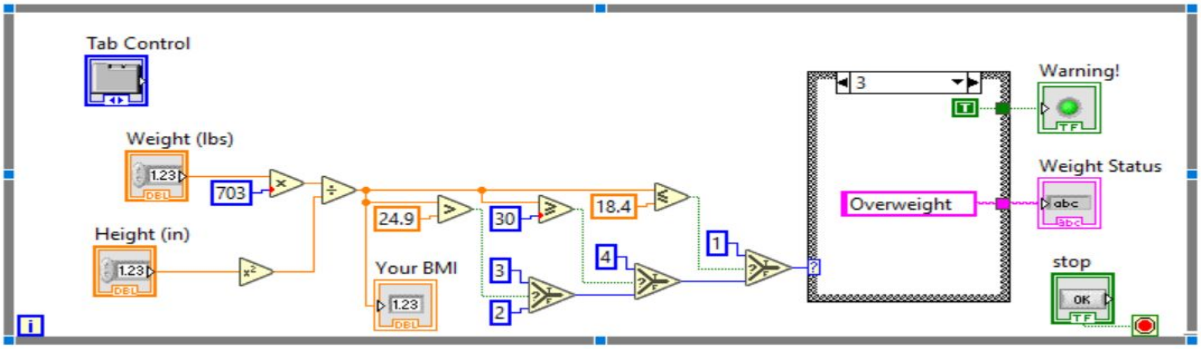
Warning!

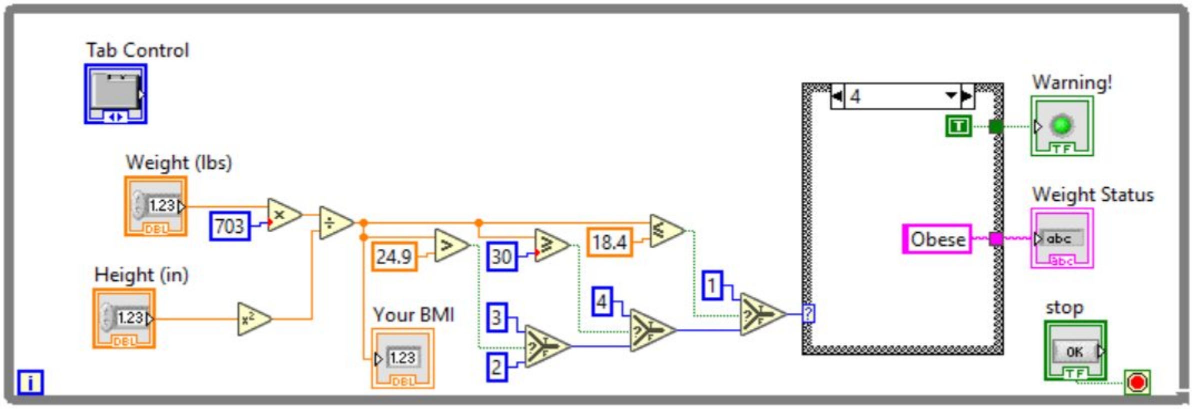
Your BMI: 23.8159

Weight Status: Healthy


Tab Control
Weight (lbs)
1.23
DBL
703
Height (in)
1.23
DBL
24.9
30
18.4
Your BMI
1.23
DBL
3
2
4
1
1, Default
T
UnderWeight
Warning!
Weight Status
abc
stop
OK

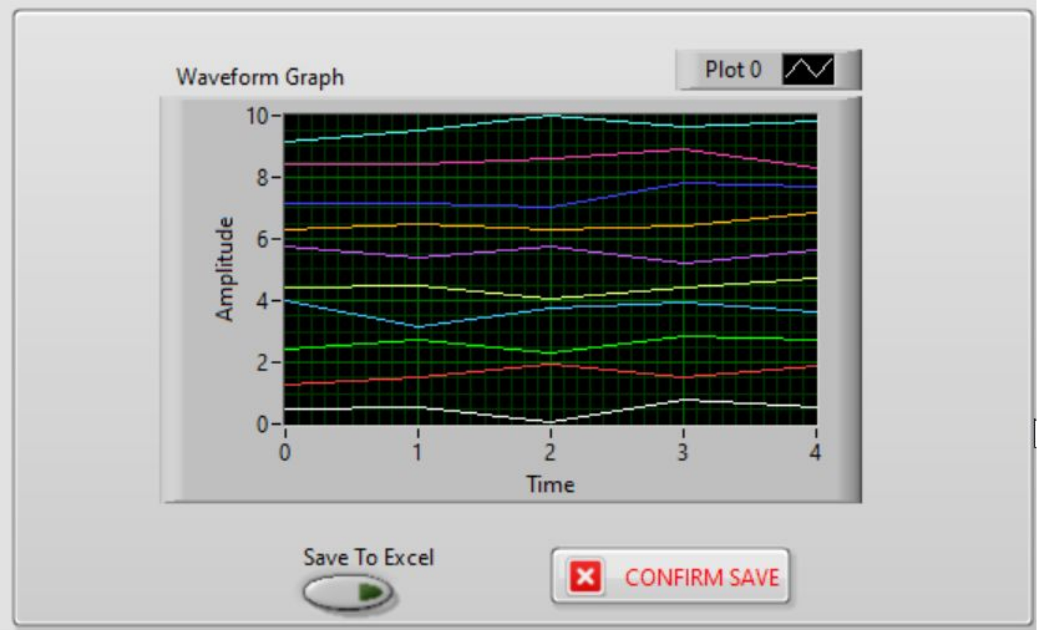

Tab Control
Weight (lbs)
1.23
DBL
703
Height (in)
1.23
DBL
24.9
30
18.4
Your BMI
1.23
DBL
3
2
4
1
2
F
Healthy
Warning!
Weight Status
abc
stop
OK

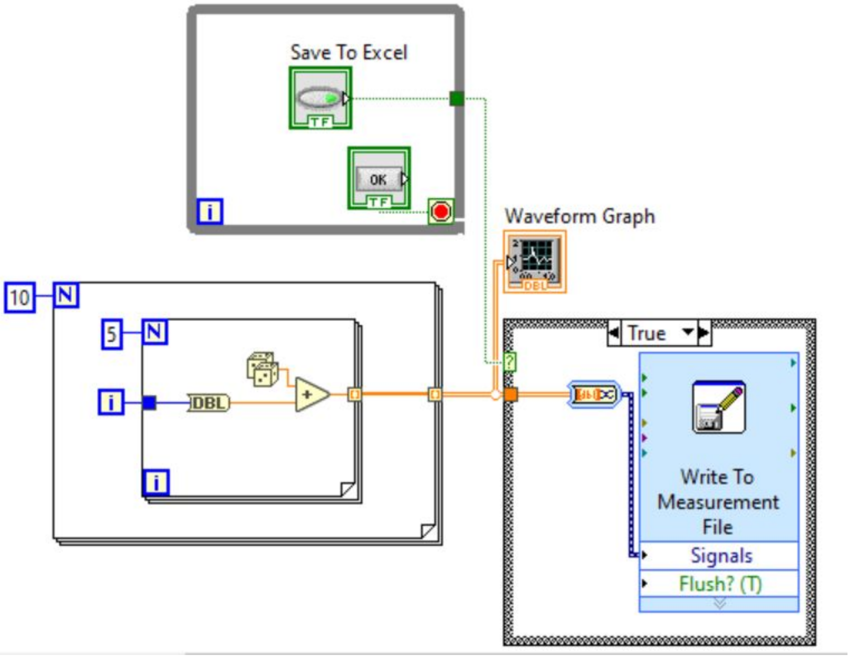

Tab Control
Weight (lbs)
1.23
DBL
703
Height (in)
1.23
DBL
24.9
Your BMI
1.23
DBL
30
3
2
4
18.4
1
3
T
Overweight
?
Warning!
TF
Weight Status
abc
stop
OK
TF
i

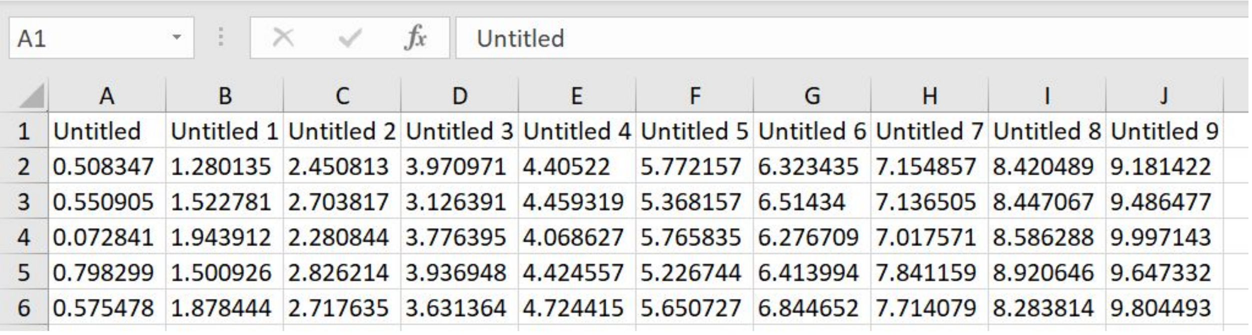

Tab Control
Weight (lbs)
1.23
DBL
703
Height (in)
1.23
DBL
24.9
Your BMI
1.23
DBL
30
3
2
4
18.4
1
4
T
Obese
?
Warning!
TF
Weight Status
abc
stop
OK
TF
i

2. **Construct a VI that outputs a 2D array with 10 rows and 5 columns.**

“Note: To save results to excel: Click the Toggle Button, then click on “Confirm Save”**.**



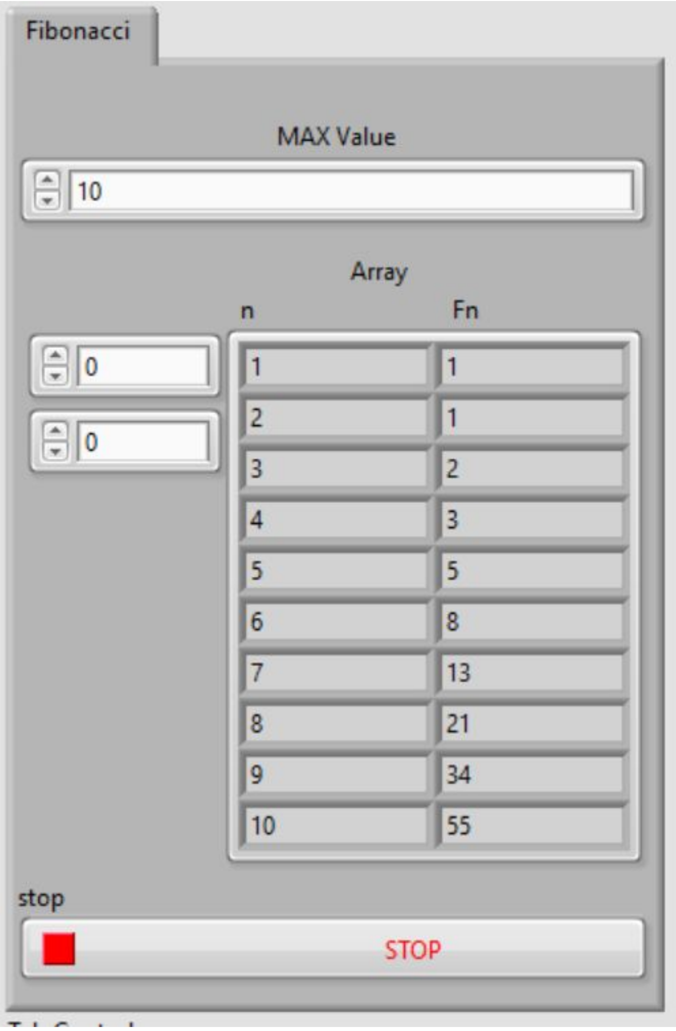




A1 Untitled

	A	B	C	D	E	F	G	H	I	J
1	Untitled	Untitled 1	Untitled 2	Untitled 3	Untitled 4	Untitled 5	Untitled 6	Untitled 7	Untitled 8	Untitled 9
2	0.508347	1.280135	2.450813	3.970971	4.40522	5.772157	6.323435	7.154857	8.420489	9.181422
3	0.550905	1.522781	2.703817	3.126391	4.459319	5.368157	6.51434	7.136505	8.447067	9.486477
4	0.072841	1.943912	2.280844	3.776395	4.068627	5.765835	6.276709	7.017571	8.586288	9.997143
5	0.798299	1.500926	2.826214	3.936948	4.424557	5.226744	6.413994	7.841159	8.920646	9.647332
6	0.575478	1.878444	2.717635	3.631364	4.724415	5.650727	6.844652	7.714079	8.283814	9.804493

3. **Construct a VI with one numeric input n that builds an array containing n Fibonacci numbers, beginning with F1. Fibonacci numbers are the sequence of numbers, denoted by Fn**.



Fibonacci

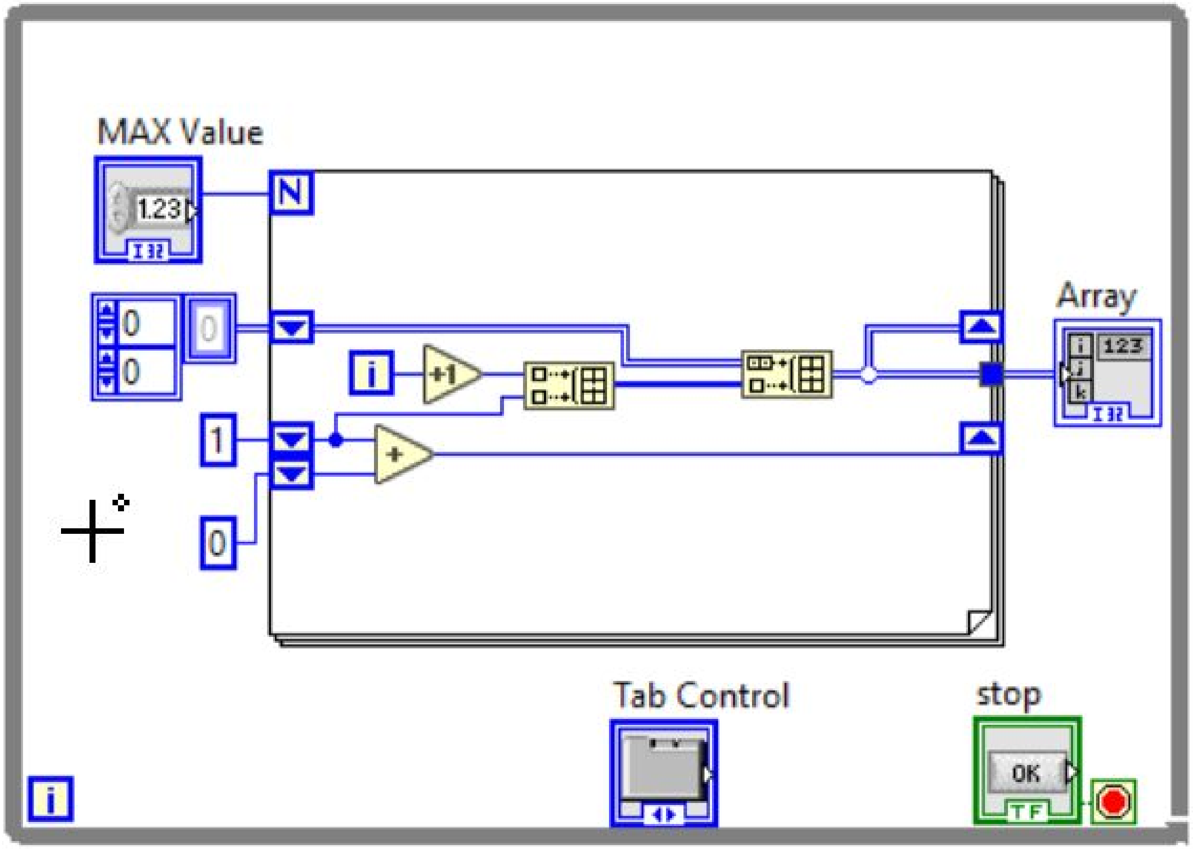
MAX Value: 10

Array

0
0

n	Fn
1	1
2	1
3	2
4	3
5	5
6	8
7	13
8	21
9	34
10	55

stop


MAX Value
1.23
I32
N
0
0
0
i
+1
1
+
0
Array
i
j
k
123
I32
Tab Control
stop
OK
TF
i