Overview. This assignment will implement a bubble sort algorithm that sorts an array in ascending order, and measures the CPU performance of this algorithm for a number of input array sizes.

Program Requirements. Your program will implement the bubble sort algorithm given below. Then you will create arrays of varying sizes, starting with 1000 integer elements, and sort the array using bubble sort. Further, your program will observe the amount of time it took to execute the sort, and print out that time.

The size of the array to be sorted should be doubled repeatedly until it reaches 512,000, which is the last array size to be sorted. In addition to the bubble sort algorithm, you must also implement a subroutine that verifies that an array is properly sorted in ascending order.

Expected Outputs. A correct solution will print the following:

```
Bubble Sort of size 1000 took xxxx milliseconds
Bubble Sort of size 2000 took xxxx milliseconds
Bubble Sort of size 4000 took xxxx milliseconds
Bubble Sort of size 8000 took xxxx milliseconds
Bubble Sort of size 16000 took xxxx milliseconds
Bubble Sort of size 32000 took xxxx milliseconds
Bubble Sort of size 64000 took xxxx milliseconds
Bubble Sort of size 128000 took xxxx milliseconds
Bubble Sort of size 256000 took xxxx milliseconds
Bubble Sort of size 512000 took xxxx milliseconds
```

In the above outputs, the “xxxx” values are the actual number of milliseconds it took to perform the sort.

Here’s a snippet of code to measure time for a piece of code

```
import datetime

# Get time before your code
#
a = datetime.datetime.now()

# Your code goes here

# Get time after your code
#
b = datetime.datetime.now()

# The next line will print how long “Your code” took
#
print(b - a)
```