CSE320 Lab Report .Python

Result for quicksort

```
[006704029@csusb.edu@csevnc lan2Python]$ python qspy.py
unsorted array:
[11, 22, 44, 33, 77, 66, 55, 99]
        Start time: 1583986183.51
        End time: 1583986183.51
        Run time: 2.50339508057e-05
sorted array:
[11, 22, 33, 44, 55, 66, 77, 99]


unsorted array:
[112, 28, 81, 198, 92, 466, 58, 597, 46, 989]
        Start time: 1583986183.51
        End time: 1583986183.51
        Run time: 2.00271606445e-05
sorted array:
[28, 46, 58, 81, 92, 112, 198, 466, 597, 989]


unsorted array:
[222, 444, 587, 456, 455, 678, 716, 729, 782, 239, 195, 495, 794, 309, 988]
        Start time: 1583986183.51
        End time: 1583986183.51
        Run time: 3.19480895996e-05
sorted array:
[195, 222, 239, 309, 444, 455, 456, 495, 587, 678, 716, 729, 782, 794, 988]
```

Result for Partition

```
[006704029@csusb.edu@csevnc CSE320]$ cd lan2Python
[006704029@csusb.edu@csevnc lan2Python]$ ls
partipy.py  py1.png  py2.png  qspy.png  qspy.py
[006704029@csusb.edu@csevnc lan2Python]$ python partipy.py
Can be divided into two subsets of equal sum
[006704029@csusb.edu@csevnc lan2Python]$ python partipy.py
successfully divided
[006704029@csusb.edu@csevnc lan2Python]$
```

Source code

```
import time
def partition (array, l, h):
    pivot = array[h]
    i = (l - 1)
    for j in range(l, h):
        if(array[j] < pivot):
            i += 1
            temp = array[i]
            array[i] = array[j]
            array[j] = temp

    temp = array[i+1]
    array[i+1] = array[h]
    array[h] = temp
    return (i+1)

def quicksort (array, l, h):
    if (l < h):
        partitionindex = partition(array, l, h)
        quicksort(array, l, partitionindex - 1)
        quicksort(array, partitionindex + 1, h)

def printtime(array, arraysize):
    print("unsorted array: ")
    print(array)
    start = time.time()
    quicksort(array, 0, arraysize -1)
    end = time.time()
    print "\tStart time:",
    print(start)
    print "\tEnd time:",
    print(end)
    print "\tRun time:",
    print(end-start)
    print("sorted array: ")
    print(array)
    print("\n")
```

```
if __name__ == "__main__":
    array1 = [11, 22, 44, 33, 77, 66, 55,99]
    arraysize1 = len(array1)
    printtime(array1, arraysize1)
    array2 = [112, 28, 81, 198, 92, 466, 58, 597, 46, 989]
    arraysize2 = len(array2)
    printtime(array2, arraysize2)
    array3 = [222, 444, 587, 456, 455, 678, 716, 729, 782, 239, 195, 495, 794, 309, 988]
    arraysize3 = len(array3)
    printtime(array3, arraysize3)
```

```python
def isSubsetSum (arr, n, sum):
    if sum == 0:
        return True
    if n == 0 and sum != 0:
        return False

    if arr[n-1] > sum:
        return isSubsetSum (arr, n-1, sum)

    return isSubsetSum (arr, n-1, sum) or isSubsetSum (arr, n-1, sum-arr[n-1])

def findPartion (arr, n):

    sum = 0
    for i in range(0, n):
        sum += arr[i]

    if sum % 2 != 0:
        return false

    return isSubsetSum (arr, n, sum // 2)

arr = [1,2,3]
n = len(arr)
if findPartion(arr, n) == True:
    print ("successfully divided")
else:
    print ("Can't be divided")
```

How easy/hard was it was to program?

In my opinion, python is an easier version of Cpp, therefore it is really easy to program.

The ease/difficulty of debugging:
The same as the question above.

The speed of execution:
The run time already shown in the screenshot.