Quiz 11

Source Code:

```
def quicSortAlgo(arr,l,a):

    size = a - l + 1
    stack = [0] * (size)

    top = -1

    top = top + 1
    stack[top] = l
    top = top + 1
    stack[top] = a

    while top >= 0:

        a = stack[top]
        top = top - 1
        l = stack[top]
        top = top - 1

        p = breakk( arr, l, a )


        if p-1 > l:
            top = top + 1
            stack[top] = l
            top = top + 1
            stack[top] = p - 1


        if p+1 < a:
            top = top + 1
            stack[top] = p + 1
            top = top + 1
            stack[top] = a


def breakk(arr,l,a):
    i = ( l - 1 )
    x = arr[a]

    for j in range(l , a):
        if arr[j] <= x:
```

```
            i = i+1
            arr[i],arr[j] = arr[j],arr[i]

    arr[i+1],arr[a] = arr[a],arr[i+1]
    return (i+1)


arr = [4, 3, 7, 55, 22, 75, 43, 18, 58, 50]
n = len(arr)
quicSortAlgo(arr, 0, n-1)
print ("Sorted array is:")
for i in range(n):
    print ("%d" %arr[i])
```

Output:

harshavaidhyam@Harshas-MacBook-Pro quiz 11 % /usr/bin/env /usr/local/bin/python3 /Users/harshavaidhyam/.vscode/extensions/ms-python.python-2022.16.1/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher 49369 -- /Users/harshavaidhyam/Desktop/Pitt\ term-1/Algo\ Design/quiz\ 11/quiz11.py

Sorted array is:
3
4
7
18
22
43
50
55
58
75

Time Complexity:

Best case and averge case:

O(n*log(n))

Worst case:

O(n^2)