Chapter 6: Sorting and Searching

Last Update: March 22, 2023

Program 1:

06 11-sorted.py

```python
def next():
    input("Next> ")

str_list = ['apple', 'banana', 'cherry', 'watermelon', 'pear']
num_list = [23, 49, -15, 8, 77]
mix_list = ['apple', 'banana', 77, 66]

print(f"str_list: {str_list}\nnum_list: {num_list}  \n")
next()
print(f"str_list: {sorted(str_list)}\nnum_list: {sorted(num_list)}
\n")
next()
print(f"str_list: {sorted(str_list, reverse=True)}\nnum_list:
{sorted(num_list, reverse=True)}  \n")
next()
print(f"str_list: {sorted(str_list, key=len)}\nnum_list:
{sorted(num_list, key=abs)}  \n")
next()
newlist = sorted(num_list, key=abs, reverse=True)
print(f"num_list sorted key=abs reverse: {newlist}")
next()

# Using sort()
str_list.sort(key=len)
print(f"sort on length of str_list: {str_list}")
next()

num_list.sort(key=abs, reverse=True)
print(f"reverse sort on absolute value of num_list: {num_list}")
next()

# This one will fail
print('mix_list:', sorted(mix_list))
```

Program 2:

06 12-sorted 2d.py

```python
def next():
    input("Next> ")

def addup(elem):
    return elem[0]+elem[1]

def on_y(elem):
    return elem[1]

def diff(elem): # absolute diff
    return abs(elem[0]-elem[1])

mylist = [(1,1), (2,-2), (-1, 0), (1,0), (1,3), (2,4)]
print(f"\nmylist: {mylist}")
next()
print(f"mylist sorted: {sorted(mylist)}")
next()
print(f"mylist sorted by addup: {sorted(mylist, key=addup)}")
next()
print(f"mylist sorted by diff:  {sorted(mylist, key=diff)}")
next()
print(f"mylist reverse sorted by on_y:  {sorted(mylist, key=on_y,
reverse=True)}")
```

Program 3:

06 13-sorted tuple.py

```python
tuple1 = (3, 5, 4, -1)
tuple2 = ((1,2), (4,5), (3,2), (2,4))
print(f"Tupke 1: {tuple1}\nTuple 2: {tuple2}\n")
print("Tuple 1 sorted in reverse", sorted(tuple1, reverse=True))
list1 = sorted(tuple1)
print("Tuple 1 sorted", list1)
print("Tuple 1 sorted", tuple(list1))
print("\nTuple 2 sorted", sorted(tuple2))
```

Program 4:

06 14-sorted.py

```python
def next():
    input("Next> ")

list1 = [19, 3, 41, 45, -22, 111, 2, 29]
list2 = [9, 8, 7, 6, 5, 4, 3, 2, 1]
list3 = ['a', 'b', 'C', 'A', 'B', 'c', 'aa', 'about', 'this', 'that']

print(list1)
print(list2)
print(list3)
next()

list4 = sorted(list1, key=abs)
print(f"List1 sorted key=abs {list4}")
next()

list5 = sorted(list2)
print(f"List2 sorted {list5}")
next()

list6 = sorted(list3, key=str.lower, reverse=True)
print(f"List3 sorted key=str.lower() reverse {list6}")
```

Program 5:

06 15-sort sorted.py

```python
def next():
    input("Next> ")

str_list = ['banana', 'apple', 'watermelon', 'pear', 'peach']
nbr_list = [10, -3, 15, 9, 99, -77, 9]
print(f"str_list = {str_list}")
print(f"nbr_list = {nbr_list}")
next()

print(f"returned by sorted(): {sorted(nbr_list)}")
print(f"returned by sorted(): {sorted(str_list, reverse=True)}\n")
next()

print(f"str_list = {str_list}")
print(f"nbr_list = {nbr_list}")
# The lists were not changed
next()

print(str_list.sort())
print(nbr_list.sort(reverse=True))
next()

print(f"\nafter sort(): str_list = {str_list}")
print(f"after sort(): nbr_list = {nbr_list}")
# The change to the lists were done through the parameters
```

Program 6:

06 21-bubble.py

```
def next():
    input("Next> ")

def bubble(a):
    for ub in range(len(a)-1, 0, -1):
        for i in range(ub):
            if a[i] > a[i+1]:
                a[i], a[i+1] = a[i+1], a[i]
        print(a)
    return(a)

list1 = [19, 3, 41, 45, 22, 111, 2, 29]
list2 = [5, 7, 4, 9, 3]
list3 = ['a', 'b', 'C', 'A', 'B', 'c', 'aa', 'about', 'this', 'that']

print(list1)
bubble(list1)
print()

print(list2)
bubble(list2)
print()

print(list3)
bubble(list3)
print()
```

Program 7:

06 22-bubble detail.py

```python
def next():
    input("Next> ")

def bubble(a):
    for ub in range(len(a)-1, 0, -1):
        print('***', ub)
        for i in range(ub):
            print(' ' * 32, f'{a[i]:3d} <> {a[i+1]:3d}', end = ' ')
            if a[i] > a[i + 1]:
                print(' <- Swap')
                a[i], a[i + 1] = a[i + 1], a[i]
            else:
                print()
        print(a)
    return(a)   # ?


list1 = [19, 3, 41, 45, 22, 111, 2, 29]
list2 = [9, 8, 7, 6, 5]
list3 = [1, 3, 5, 7, 9, 11]

print(list1)
list4 = bubble(list1)
print(list1)
next()

print(list2)
list2 = bubble(list2)
next()

print(list3)
list3 = bubble(list3)
```

Program 8:

06 23-selection sort.py

```

def next():
    input("Next> ")

def swap(list, i, j):
    list[i], list[j] = list[j], list[i]

def selection_sort(a):
    for i in range(len(a)-1):
        min_i = i
        for j in range(i +1, len(a)):
            if a[min_i] > a[j]:
                min_i = j
        swap(a, i, min_i)
        print(a)


list1 = [19,2,31,45,30,11,121,27]
print(list1)
selection_sort(list1)
next()

list2 = [99,88,77,66, 55]
print(list2)
selection_sort(list2)
next()

list3 = [8, 7, 6, 5, 4, 3, 2, 1]
print(list3)
selection_sort(list3)
next()

list3 = [6, 5, 7, 8, 4, 3, 2, 1]
print(list3)
selection_sort(list3)
```

Program 9:

06 24-selection detail.py

```
def next():
    input("Next> ")

def swap(list, i, j):
    list[i], list[j] = list[j], list[i]

def selection_sort(a):
    for i in range(len(a)-1):
        min_i = i
        for j in range(i +1, len(a)):
            if a[min_i] > a[j]:
                min_i = j
        swap(a, i, min_i)
        print(a)


list1 = [19,2,31,45,30,11,121,27]
print(list1)
selection_sort(list1)
next()

list2 = [99,88,77,66, 55]
print(list2)
selection_sort(list2)
next()

list3 = [8, 7, 6, 5, 4, 3, 2, 1]
print(list3)
selection_sort(list3)
next()

list3 = [6, 5, 7, 8, 4, 3, 2, 1]
print(list3)
selection_sort(list3)
```

Program 10:

```
06 31-linear.py
def next():
    input("Next> ")

def search(x, a):
    foundIndex= None
    count = 0
    for i in range(len(a)):
        if a[i]==x:
            count += 1
            foundIndex=i
    return foundIndex, count

mylist = [2, 3, 25, 7, 7, 4, 9, 8, 7, 16, 25, 2]
test = [10, 8, 7]
for j in test:
    print('\nList:', mylist)
    idx, count = search(j, mylist)
    if idx != None:
        print(f"Searching for {j}: Found, index = {idx}, value =
{mylist[idx]}, count = {count}")
    else:
        print(f"Searching for {j}: Not found, count = {count}")
    next()
```

Program 11:

```
06 32-linear-first.py
def next():
    input("Next> ")

def search(x,a):
    count = 0
    for i in range(len(a)):
        if a[i]==x:
            count += 1
            return i, count
    return None, count

mylist = [2, 3, 25, 7, 7, 4, 9, 8, 7, 16, 25]
test = [10, 8, 7]

for j in test:
    print('\nList:', mylist)
    idx, count = search(j, mylist)
    if idx!=None:
        print(f"Searching for {j}: Found, index = {idx}, value =
{mylist[idx]}, count = {count}")
    else:
        print(f"Searching {j}: Not found.  count =  {count}")
    next()
```

Program 12:

06 33-recursive.py

```python
def search(x,list,lb,ub):
    print(f"Searching {x} between {lb} and {ub}")
    if lb >= ub:
        return None
    elif list[lb]==x:
        return lb
    else:
        return(search(x,list,lb+1, ub))

mylist = [2, 3, 25, 7, 7, 4, 9, 8, 7, 16, 25]
test = [10, 8, 7]
for j in test:
    print('\nList:', mylist)
    idx = search(j, mylist, 0, len(mylist))
    if idx != None:
        print(f"Searching for {j}: Found, index = {idx}, value = {mylist[idx]}.")
    else:
        print(f"Searching for {j}: Not found.")
```

Program 13:

06 41-binary.py

```python
def search(x,a):
    low, high = 0, len(a)
    while low<high:
        mid=(low+high)//2
        print('\n', a[low:high])
        print(f"Search Range [{low:2d},{high:2d}), mid = {mid:2d},", end=' ')
        if x==a[mid]:
            print('=', f'{a[mid]:2d}', end=' ')
            return mid
        elif x<a[mid]:
            print('<', f'{a[mid]:2d}', end=' ')
            high=mid
        else:
            print('>', f'{a[mid]:2d}', end=' ')
            low =mid+1
        print(f"New Range [{low:2d},{high:2d})")
    return None

a = [3, 5, 7, 9, 10, 11, 12, 13, 14, 15, 17, 23, 32, 35, 37, 43, 55, 66, 77]
print(a)
x = int(input('Enter an integer to search: '))
idx = search(x, a)
if idx!=None:
    print(f"\nFound {x} at index = {idx}")
else:
    print(f"\n{x} not found")
```

Program 14:

06 42-binary.py

```python
def search(x,a):
    low, high = 0, len(a)
    while low<high:
        mid=(low+high)//2
        print('\n', a[low:high])
        print(f"Search Range [{low:2d},{high:2d}), mid = {mid:2d},",
end=' ')
        if x==a[mid]:
            print('=', f'{a[mid]:2d}', end=' ')
            return mid
        elif x<a[mid]:
            print('<', f'{a[mid]:2d}', end=' ')
            high=mid
        else:
            print('>', f'{a[mid]:2d}', end=' ')
            low =mid+1
        print(f"New Range [{low:2d},{high:2d})")
    return None

a = [3, 5, 7, 9, 10, 11, 12, 13, 14, 15, 17, 23, 32, 35, 37, 43, 55,
66, 77]
print(a)
x = int(input('Enter an integer to search: '))
idx = search(x, a)
if idx!=None:
    print(f"\nFound {x} at index = {idx}")
else:
    print(f"\n{x} not found")
```

Program 15:

06 43-recursive binary.py

```python
def b_search(x,list, low, high):
    while low<=high:
        mid=(low+high)//2
        print("b_search(",x,",list,",low, ",",high,")", sep='')
        if x==list[mid]:  return mid
        elif x<list[mid]: return(b_search(x,list,low,mid-1))
        else:             return(b_search(x,list,mid+1,high))
    return None

a = [3, 5, 7, 9, 10, 12, 15, 17, 23, 32, 35, 37, 43]

print (a)
idx = b_search(10, a, 0, len(a))
if idx!=None:
    print("Found index =", idx, "value = ", a[idx])
else:
    print("Not found")
```