

**BRONX COMMUNITY COLLEGE**  
**of The City University of New York**

**DEPARTMENT OF MATHEMATICS and COMPUTER SCIENCE**

**CSI 33**

**Midterm Exam Sample/outline with Answers**

**6.** The integers 20, 12, 7, 14, 2, 5, 3 and 8 are inserted in that order into a container object. Give the order in which these values are retrieved, if the container is

1. a stack,

**Answer:** Stack is LIFO (Last In First Out), therefore the 8 will be retrieved first, then 3 and so forth. The elements will be retrieved in the following order: 8, 3, 5, 2, 14, 7, 12, 20

2. a queue.

**Answer:** Queue is FIFO (First In First Out), therefore since 20 was the first element to go into queue, it will be the first element to retrieve.

The elements will be retrieved in the following order: 20, 12, 7, 14, 2, 5, 3, 8

**9.**

```
def f1(a,b,c):
    a.append(b)
    a.append(c)
    a = b+c
    b += 10
    print(a,b)
    return a

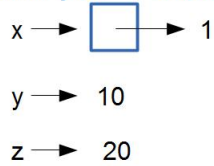
def main():
    x=[1]
    y,z = 10,20
    y = f1(x,y,z)
    print(x,y,z)
```

**Answer:**

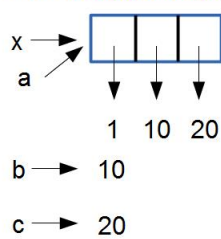
In the Python shell will be displayed the following:

```
30 20
[1,10,20] 30 20
```

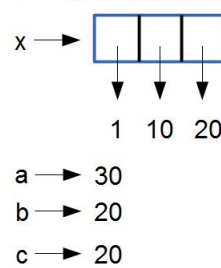
In the main() function,  
after `y, z = 10, 20`:



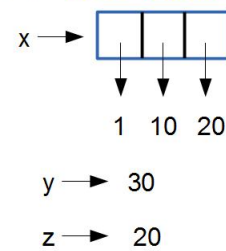
In the f1() function,  
after `a.append(c)`:



In the f1() function,  
after `print(a, b)`:



In the main() function,  
after `print(x, y, z)`:



Here is what will be happening in memory:

13 (a) see in-class work

13 (b)

```
def binStrings(n):
    """ pre: n is positive integer
        post: returns the list of all possible binary strings of length n
        recursive function"""

    result = []

    if n == 0:
        return ['']

    else:

        resultIntermediate = binStrings(n-1)

        for item in resultIntermediate:
            result.append(item+'0')
            result.append(item+'1')

    return result

print(binStrings(2))
print(binStrings(5))
```