





9. (3 pts) Write a function that takes two numbers and returns the largest.

10. (4 pts) Write code that defines a class called Cat. Give it one attribute and one method.

11. (3 pts) Write code that creates an instance of Cat. Set the attribute and call the method.

12. (3 pts) Write Python code that will create an array of 100 zeros.

13. (2 pts) Write code to print the first and last values:

```
list = [55, 41, 52, 68, 45, 27, 40, 25, 37, 26]
```

14. (1 pt) Write the code that would print the cell that contains the number 1 given the following grid of numbers:

	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	0	0
2	0	1	0	0	0
3	0	0	0	0	2
4	0	0	0	0	0

15. (3 pts) Write code to swap the values 41 and 55. (3 lines of code)  
list = [55, 41, 52, 68, 45, 27, 40, 25, 37, 26]

16. (1 pt) Under what circumstances would a linear search work well, but a binary search would not work at all?

17. (3 pts) Show how to perform a selection sort on the following numbers:

9                    3                    4                    1                    0                    2

18. (3 pts) Show how to perform a insertion sort on the following numbers:

9                    3                    4                    1                    0                    2

19. (3 pts) Write code that will take input from a user, and then convert it to a number. Print an error if the conversion fails.

20. (4 pts) Take the following code and fill in the blanks:

```
def insertion_sort(list):  
  
    # Start at the second element (pos 1).  
    # Use this element to insert into the  
    # list.  
    for keyPos in range(1, len(list)):  
  
        # Get the value of the element to insert  
        keyValue = _____  
  
        # Scan to the left  
        scanPos = keyPos - 1  
  
        # Loop each element, moving them up until  
        # we reach the position  
  
        while (scanPos >=0) and ( _____ ):  
            list[scanPos+1] = list[scanPos]  
            _____  
  
        # Everything's been moved out of the way, insert  
        # the key into the correct location  
        list[scanPos+1] = _____
```

21. (1 pt) Explain what minPos does in the selection sort.

22. (5 pts) Take the following code and fill in the blanks:

```
# The selection sort
def selection_sort(list):

    # Loop through the entire array

    for curPos in range( _____ ):

        # Find the position that has the smallest number
        # Start with the current position

        minPos = _____

        # Scan right
        for scanPos in range(curPos+1, _____ ):

            # Is this position smallest?
            if list[scanPos] < list[minPos]:

                # It is, mark this position as the smallest
                minPos = scanPos

        # Swap the two values
        temp = list [ minPos ]

        _____

        _____
```

23. (1 pt) Explain what keyPos and keyValue are in the insertion sort.



24. Write code to format the following variables:

```
amount1=1.1  
amount2=12.02  
total=amount1+amount2
```

So they print like the following:

```
Amount 1:    $  1.10  
Amount 2:    $ 12.02  
Total:       $ 13.12
```

25. Explain the relationship between Microsoft and IBM in the mid 1980's:

26. Where did Apple get the idea for the GUI?

27. Explain the relationship between John Sculley and Steve Jobs:

28. What is ARPANet?

29. What is packet switching?