```
class Student:
        def __init__(self, name, id):
                self.__name = name
                self.__id = id
                self.__testScores = []

        def addScoreToArray(self,score):
                if score >= 0:
                        self.__testScores.append(score)

        def displayAllScores(self):
                print(self.__name," ",self.__id)

                for i in range(len(self.__testScores)):
                        print(i+1,": ",self.__testScores[i])

        def getACount(self):
                #your code

        def sendScoresToFile(self):
                #your code


s = Student("Jack", "s01111")

print("Enter a score or -1 to quit:")

score = int(input())

while score != -1:
        s.addScoreToArray(score)
        s.displayAllScores()
        print("Enter a score or -1 to quit")
        score = int(input())

count = s.getACount()
print("You have achieved ",count," A grades")

s.sendScoresToFile()
```

A student object stores 3 attributes.
A student has scores appended to an array.
All scores can be displayed.

A method displays a count of all scores above 89%

A method opens a file having the same name as the student's id eg s01111.txt and writes each score in the array into the file, one score per line, including a line number eg:

1. 89
2. 67
3. 78

Find the line

Circle the line of code which appends an INTEGER to the ARRAY.

Circle a private data attribute.

Circle the line of code which constructs a Student object.

Circle a line of code which calls a Student class method.

Adaptive Maintenance

The above program has no syntax errors, no logic errors and no runtime errors.

Type up the program and then perform some adaptive maintenance and implement the two methods that have not yet been coded.

Once you have implemented the displayACount method and tested/fixed it, continue with the sendScoresToFile method. You may want to comment out or delete unnecessary code above as you are designing your solution.