

Midterm Exam Version 1 (120 points)

Question 1: (10 pts) Show the output from the following code segments:

	Code	Output												
<p>A (5)</p>	<pre> int [] numsA = {20, 15, 11, 22, 75}; int [] numsB = {2, 4, 6, 8, 10, 12}; numsB[0] = numsA[1]; numsB[3] = numsB[1] * numsA[0]; numsA[0] = 50; numsA = numsB; if(numsA[5] == 12){     System.out.println("trick"); } else{     System.out.println("treat"); } System.out.println("A : "+Arrays.toString(numsA)); System.out.println("B : "+Arrays.toString(numsB));                 </pre>													
<p>B (5)</p>	<pre> public static void q1B(char c, int x){     switch(c){         case 'a':         case 'A':             if(x % 2 == 0){                 System.out.print("sweet ");             }             System.out.print("candy ");         case 'b':             System.out.print("berries ");         case 'c':             System.out.println("corn");             break;         case 'd':             System.out.println("eaten");             while(x&gt;0){                 System.out.println("by monsters");                 x--;             }             break;         default:             System.out.println("parrots");             Break;     } }                 </pre>	<table border="1"> <thead> <tr> <th data-bbox="787 919 1015 1035">method call</th> <th data-bbox="1015 919 1542 1035">output</th> </tr> </thead> <tbody> <tr> <td data-bbox="787 1035 1015 1207">q1B('a', 3);</td> <td data-bbox="1015 1035 1542 1207"></td> </tr> <tr> <td data-bbox="787 1207 1015 1379">q1B('b', 2);</td> <td data-bbox="1015 1207 1542 1379"></td> </tr> <tr> <td data-bbox="787 1379 1015 1551">q1B('c', 4);</td> <td data-bbox="1015 1379 1542 1551"></td> </tr> <tr> <td data-bbox="787 1551 1015 1724">q1B('d', 3);</td> <td data-bbox="1015 1551 1542 1724"></td> </tr> <tr> <td data-bbox="787 1724 1015 1896">q1B('e', 7);</td> <td data-bbox="1015 1724 1542 1896"></td> </tr> </tbody> </table>	method call	output	q1B('a', 3);		q1B('b', 2);		q1B('c', 4);		q1B('d', 3);		q1B('e', 7);	
method call	output													
q1B('a', 3);														
q1B('b', 2);														
q1B('c', 4);														
q1B('d', 3);														
q1B('e', 7);														

**Midterm Exam Version 1 (120 points)**

**Question 2: (36 pts)** Write a complete class **Vehicle** that has the following attributes, constructors and methods. Use the UML and explanations below.

Vehicle

- numVehicles : static int //initialize to 0, increment inside the constructor
- model : String
- vinNumber : int
- # generateVinNumber() : void //method to increment numVehicles and create the vinNumber
- + Vehicle() //increment numVehicles and assign value to vinNumber
- + Vehicle(String) //Call the default Vehicle constructor and then set the model
- + getNumVehicles() : static int //no setter for numVehicles
- + getVinNumber() : int //no setter for vinNumber
- + getModel() : String
- + setModel(String) : void
- + toString() : String
- + equals() : boolean

2a) (3) Write all **variables** as per the UML

2b) (12) Write all the **constructors** as per the UML

Note: Do not call the **generateVinNumber** method from within the Vehicle constructor

2c) (3) Write the **generateVinNumber** method

Note: Used to generate a replacement vinNumber, and is available for inheritance

2d) (8) Write the **getters/accessors** and **setters/mutators** as per the UML

2e) (10) Write the overridden methods from the Object class (**toString** and **equals**)

Note1: The **toString** should be formatted as "Vehicle: Model= %20s | VIN= %10d\n".

Note2: The **equals** method should use the **model** and **vinNumber** to determine equality.

**Midterm Exam Version 1 (120 points)**

A large, empty rectangular box with a thin black border, occupying most of the page below the header. It is intended for the student to write their answers to the exam questions.

**Midterm Exam Version 1 (120 points)**

**Question 3: (4 pts)** Write a class named **CargoCapacityException** that inherits from **Exception**. Make sure to define both the **CargoCapacityException()** and **CargoCapacityException(String)** constructors.

---

Use the following interface when answering questions 4 and 5.

```
public interface Movable {  
    void move();  
    void park();  
}
```

---

**Midterm Exam Version 1 (120 points)**

**Question 4: (15 pts)** Write a complete class **Car** that extends the **Vehicle** class and implements the **Movable** interface provided. Use the UML and explanations below.

Car

- cId : String // is the inherited int vinNumber prefixed by the String "C" as shown "C" + vinNumber
- numDrivers : int
- driverNames [ ] : String
- + toString() : String

**NOTE:** Assume all other necessary constructors, getters, setters, and other methods have been written already even though they are not shown here (**do not rewrite them**)

**4a) (3)** Write a complete class **Car** that extends the **Vehicle** class and implements the **Movable** interface provided.

**4b) (3)** Write all **variables** as per the UML shown above

**4c) (3)** Write one of the overridden methods from the **Vehicle** class (**toString**)

Note: Remember to include the functionality of the parent class's toString() method. In addition to parent class's functionality, the output should have:

"Car: cId= %10s | numDrivers= %10d\n".

**4d) (3)** Implement the **move()**; method from the **Movable** interface so that it prints out "Hit the gas!"

**4e) (3)** Implement the **park()** method from the **Movable** interface so that it prints out "Squeezing between two other cars"

**Midterm Exam Version 1 (120 points)**

A large, empty rectangular box with a thin black border, occupying most of the page below the header. It is intended for the student to write their answers during the exam.

**Midterm Exam Version 1 (120 points)**

**Question 5: (44 pts)** Write a complete class **Ship** that extends the **Vehicle** class and implements the **Movable** interface provided. Use the UML and explanations below.

Ship

- sld : String // is the inherited int vinNumber prefixed by the String "S" as shown "S" + vinNumber
- numCars : int // the current number of cars in Car[ ] array
- cabinCrewNames [ ] : String //default length of 4
- cars [ ] : Car //default and max length of 15
- # generateShipId() : void //call parent's getVinNumber() and prefix it with "S" to assign to sld "S" + VinNumber
- + Ship(String, String [ ])
- + Ship(String, String [ ], Car [ ])
- + addCar(Car) : void throws CargoCapacityException
- + getNumCars() : int
- + getCarsAsString() : String // single String, containing each Car's detail [ space separated ]
- + getSld() : String

**5a) (3)** Write all **variables** as per the UML

**5b) (18)** Write all the **constructors** and the **generateShipId** method as per the UML

Note1: Remember the **generateVinNumber** method from the **Vehicle** class is needed

Note2: If the passed in Car array is longer than 15, copy only the first 15 cars

**5c) (7)** Write the **getters/accessors** and **setters/mutators** as per the UML

**5d) (10)** Write the **addCar** method so that it adds the Car to the array and increments **numCars**. If the capacity of 15 is exceeded throw the **CargoCapacityException** instead.

**5e) (3)** Define the **move()** method from the **Movable** interface so that it prints out "Full Steam ahead!"

**5f) (3)** Define the **park();** method from the **Movable** interface so that it prints out "Carefully approach dock"

**Midterm Exam Version 1 (120 points)**

A large, empty rectangular box with a thin black border, occupying the majority of the page below the header. It is intended for the student to write their answers to the exam questions.



**Midterm Exam Version 1 (120 points)**

A large, empty rectangular box with a thin black border, occupying most of the page below the header. It is intended for the student to write their answers to the exam questions.

**Midterm Exam Version 1 (120 points)**

**Question 6: (11 pts)** Write the code to create instances of Vehicle, Ship, Car and invoke their methods as specified below.

a	Vehicle <b>v1</b> without specifying the model
b	Vehicle <b>v2</b> whose model is "Freighter"
c	Ship <b>s1</b> whose model is "Freighter" and crewMembers "Ahab" and "Popeye"
d	Ship <b>s2</b> whose model is "Container" and crewMembers "Capt. America" and "Hulk"
e	Car <b>c1</b> whose model is "Tesla" and has driverNames "Fakhouri" and "Odysseas"
f	Show the invocation of <b>toString()</b> on <b>v1</b>
g	Show the invocation that would check the if <b>v1</b> and <b>v2</b> are <b>equal</b>
h	Invoke the <b>getCarsAsString()</b> method on <b>s2</b>
i	Invoke the <b>park()</b> method on <b>s1</b>
j	Invoke the <b>park()</b> method on <b>c1</b>
k	Invoke the <b>addCar(Car)</b> method on <b>s1</b> passing in <b>c1</b> and using exception handling

--