AP® CollegeBoard

2020 Exam Sample Questions

AP® COMPUTER SCIENCE A

2020 Exam Sample Question 1

(Adapted from: AP® Computer Science A Course and Exam Description)

Directions: SHOW ALL YOUR WORK. REMEMBER THAT PROGRAM SEGMENTS ARE TO BE WRITTEN IN JAVA.

Notes:

- Assume that the classes listed in the Java Quick Reference have been imported where appropriate.
- Unless otherwise noted in the question, assume that parameters in method calls are not `null` and that methods are called only when their preconditions are satisfied.
- In writing solutions for each question, you may use any of the accessible methods that are listed in classes defined in that question. Writing significant amounts of code that can be replaced by a call to one of these methods will not receive full credit.

Question 1 - Array/ArrayList

Allotted time: 25 minutes (plus 5 minutes to submit)

The `Gizmo` class represents `gadgets` that people purchase. Some `Gizmo` objects are electronic and others are not. A partial definition of the `Gizmo` class is shown below.

```
public class Gizmo
{
    /** Returns the name of the manufacturer of this Gizmo. */
    public String getMaker()
    {   /* implementation not shown */   }
    /** Returns true if this Gizmo is electronic, and false otherwise.
     */
    public boolean isElectronic()
    {   /* implementation not shown */   }
    /** Returns true if this Gizmo is equivalent to the Gizmo object
     *  represented by the parameter, and false otherwise.
     */
    public boolean equals(Object other)
    {   /* implementation not shown */   }
    // There may be instance variables, constructors, and methods not shown.
}
```

The `OnlinePurchaseManager` class manages a sequence of `Gizmo` objects that an individual has purchased from an online vendor. You will write two methods of the `OnlinePurchaseManager` class. A partial definition of the `OnlinePurchaseManager` class is shown below.

```
public class OnlinePurchaseManager
{
    /** An ArrayList of purchased Gizmo objects, instantiated in the constructor. */
    private ArrayList<Gizmo> purchases;

    /** Returns the number of purchased Gizmo objects that are electronic and are
     *  manufactured by maker, as described in part (a).
     */
    public int countElectronicsByMaker(String maker)
    {    /* to be implemented in part (a) */    }

    /** Returns true if any pair of adjacent purchased Gizmo objects are equivalent, and
     *  false otherwise, as described in part (b).
     */
    public boolean hasAdjacentEqualPair()
    {    /* to be implemented in part (b) */    }

    // There may be instance variables, constructors, and methods not shown.
}
```

(a) Write the `countElectronicsByMaker` method. The method examines the `ArrayList` instance variable `purchases` to determine how many `Gizmo` objects purchased are electronic and are manufactured by `maker`.

Assume that the `OnlinePurchaseManager` object `opm` has been declared and initialized so that the `ArrayList purchases` contains `Gizmo` objects as represented in the following table.

Index in `purchases`	0	1	2	3	4	5
Value returned by method call `isElectronic()`	`true`	`false`	`true`	`false`	`true`	`false`
Value returned by method call `getMaker()`	`"ABC"`	`"ABC"`	`"XYZ"`	`"lmnop"`	`"ABC"`	`"ABC"`

The following table shows the value returned by some calls to `countElectronicsByMaker`.

Method Call	**Return Value**
`opm.countElectronicsByMaker("ABC")`	2
`opm.countElectronicsByMaker("lmnop")`	0
`opm.countElectronicsByMaker("XYZ")`	1
`opm.countElectronicsByMaker("QRP")`	0

Complete method `countElectronicsByMaker` below.

```
/** Returns the number of purchased Gizmo objects that are electronic and
 *  whose manufacturer is maker, as described in part (a).
 */
public int countElectronicsByMaker(String maker)
```

(b) When purchasing items online, users occasionally purchase two identical items in rapid succession without intending to do so (e.g., by clicking a purchase button twice). A vendor may want to check a user's purchase history to detect such occurrences and request confirmation.

Write the `hasAdjacentEqualPair` method. The method detects whether two adjacent `Gizmo` objects in `purchases` are equivalent, using the `equals` method of the `Gizmo` class. If an adjacent equivalent pair is found, the `hasAdjacentEqualPair` method returns `true`. If no such pair is found, or if `purchases` has fewer than two elements, the method returns `false`.

Complete method `hasAdjacentEqualPair` below.

```
/**  Returns true if any pair of adjacent purchased Gizmo objects are equivalent, and
 *   false otherwise, as described in part (b).
 */
public boolean hasAdjacentEqualPair()
```

(c) A programmer would like to add a method `getCheapestGizmoByMaker`, which returns the least expensive `Gizmo` purchased by an individual from a given maker.

Write a description of how you would change the `Gizmo` and `OnlinePurchaseManager` classes in order to support this modification.

Make sure to include the following in your response.

- Write the method header for the `getCheapestGizmoByMaker` method.
- Identify any new or modified variables, constructors, or methods aside from the `getCheapestGizmoByMaker` method. **Do not write the program code for this change.**
- Describe, for each new or revised variable, constructor, or method, how it would change or be implemented, including visibility and type. You do not need to describe the `getCheapestGizmoByMaker` method. **Do not write the program code for this change.**

2020 Exam Sample Question 2

(Adapted from: AP Computer Science A Course and Exam Description)

Directions: SHOW ALL YOUR WORK. REMEMBER THAT PROGRAM SEGMENTS ARE TO BE WRITTEN IN JAVA.

Notes:

- Assume that the classes listed in the Java Quick Reference have been imported where appropriate.
- Unless otherwise noted in the question, assume that parameters in method calls are not `null` and that methods are called only when their preconditions are satisfied.
- In writing solutions for each question, you may use any of the accessible methods that are listed in classes defined in that question. Writing significant amounts of code that can be replaced by a call to one of these methods will not receive full credit.

Question 2 – Methods and Control Structures

Allotted time: 15 minutes (plus 5 minutes to submit)

This question involves the use of check digits, which can be used to help detect if an error has occurred when a number is entered or transmitted electronically.

The `CheckDigit` class is shown below. You will write one method of the `CheckDigit` class.

```
public class CheckDigit
{
    /** Returns the check digit for num
     *   Precondition: The number of digits in num is between one and six, inclusive.
     *                 num >= 0
     */
    public static int getCheck(int num)
    {   /* implementation not shown */    }
    /** Returns true if numWithCheckDigit is valid, or false otherwise,
   *   as described in part (a)
     *  Precondition: The number of digits in numWithCheckDigit is
     *                between two and seven, inclusive.
     *                numWithCheckDigit >= 0
     */
    public static boolean isValid(int numWithCheckDigit)
    {   /*  to be implemented in part (a)  */    }
    //  There may be variables and methods not shown.
}
```

(a) Write the `isValid` method. The method returns `true` if its parameter `numWithCheckDigit`, which represents a number containing a check digit, is valid, and `false` otherwise. The check digit is always the rightmost digit of `numWithCheckDigit`.

The following table shows some examples of the use of `isValid`.

Method Call	Return Value	Explanation
`getCheck(159)`	`2`	The check digit for `159` is `2`.
`isValid(1592)`	`true`	The number `1592` is a valid combination of a number (`159`) and its check digit (`2`).
`isValid(1593)`	`false`	The number `1593` is not a valid combination of a number (`159`) and its check digit (`3`) because `2` is the check digit for `159`.

Complete method `isValid` below. You must use `getCheck` appropriately to receive full credit.

```
/** Returns true if numWithCheckDigit is valid, or false
 *  otherwise, as described in part (a)
 *  Precondition: The number of digits in numWithCheckDigit is
 *                between two and seven, inclusive.
 *                numWithCheckDigit >= 0
 */
public static boolean isValid(int numWithCheckDigit)
```

(b) A programmer wants to modify the `CheckDigit` class to keep track of how many times a call to `isValid` is made with an incorrect check digit. Any time a call to `isValid` is made with an incorrect check digit, the count should be increased by one. The programmer would like to implement this change without making any changes to the signature of the `isValid` method or overloading `isValid`.

Write a description of how you would change the `CheckDigit` class in order to support this modification. **Do not write the program code for this change.**

Make sure to include the following in your response.

- Identify any new or modified variables or methods.
- Describe, for each new or revised variable or method, how it would change or be implemented, including visibility and type.

Preparing for the AP CSA Open-ended Questions

A single paragraph (2 – 4 sentences) may be sufficient for answering these questions. The provided bullets in the question are meant to guide your single paragraph response; it is not necessary to provide a separate response for each bullet.

The following verbs are in addition to the Task Verbs in Free-Response Questions on page 190 of the **AP Computer Science A Course and Exam Description**:

- **Identify**: Provide a name for the specific addition or modification as it relates to the problem, without elaboration or explanation. For example, "I need a new variable to represent the item that appears the greatest number of times in the list."
- **Describe**: Provide the relevant features or characteristics (including any visibility or type) of your proposed modifications. For example, "I need a new instance variable, which will be a new private String object in the Sample class representing the item that appears the greatest number of times in the list." This example includes the identification and description necessary for this specific addition.

Note that these examples are not responses to the sample questions given but highlight what is expected when students are asked to identify or describe.

For features and characteristics of variables, constructors, and methods, see the following topics in the **AP Computer Science A Course and Exam Description**:

- Variables: 1.2, 2.2, 5.7, 5.8
- Constructors: 2.2, 5.2
- Methods: 5.4, 5.5, 5.6, 5.7, 5.8