IT Skill Test GIC Myanmar

Duration: 30 Minutes Total Questions: 8

1. You're developing a simple text processing utility. Which of the following code snippets correctly counts the number of words in a given string, assuming words are separated by single spaces?

   A.public int countWords(String text) { return text.split(" ").length; }

   B.public int countWords(String text) { return text.trim().split(" ").length; }

   C.public int countWords(String text) { String[] words = text.split(" "); return words.length - 1; }

   D.public int countWords(String text) { return text.split("¥w+").length; }

2. You're working on a simple game where a player's score determines their rank. The ranks are as follows: - 0-50 points: Bronze - 51-100 points: Silver - 101-150 points: Gold - Above 150 points: Platinum Which conditional logic structure would be most efficient to implement this ranking system?

   A.if (score <= 50) rank = "Bronze"; else if (score <= 100) rank = "Silver"; else if (score <= 150) rank = "Gold"; else rank = "Platinum";

   B.switch (score) { case 0-50: rank = "Bronze"; break; case 51-100: rank = "Silver"; break; case 101-150: rank = "Gold"; break; default: rank = "Platinum"; }

   C.while (true) { if (score <= 50) { rank = "Bronze"; break; } else if (score <= 100) { rank = "Silver"; break; } else if (score <= 150) { rank = "Gold"; break; } else { rank = "Platinum"; break; } }

   D.if (score > 150) rank = "Platinum"; else if (score > 100) rank = "Gold"; else if (score > 50) rank = "Silver"; else rank = "Bronze";

3. In a network communication scenario, you need to implement a retry mechanism that attempts to establish a connection up to a maximum number of times. Which control structure would be most appropriate?

   A.if statement

   B.switch statement

   C.while loop

   D.for loop

4. You're working on a Java application that processes large amounts of data. The following method is meant to filter a list of integers and return only the even numbers. However, it's causing an OutOfMemoryError when processing very large lists. Identify the most efficient fix for this issue: public List<Integer> filterEvenNumbers(List<Integer> numbers) { List<Integer> result = new ArrayList<>(); for (Integer num : numbers) { if (num % 2 == 0) { result.add(num); } } return result; }

   A.public List<Integer> filterEvenNumbers(List<Integer> numbers) { return numbers.parallelStream() .filter(num -> num % 2 == 0) .collect(Collectors.toList()); }

   B.public List<Integer> filterEvenNumbers(List<Integer> numbers) { return numbers.stream() .filter(num -> num % 2 == 0) .collect(Collectors.toList()); }

   C.public Stream<Integer> filterEvenNumbers(List<Integer> numbers) { return numbers.stream() .filter(num -> num % 2 == 0); }

   D.public List<Integer> filterEvenNumbers(List<Integer> numbers) { List<Integer> result = new LinkedList<>(); for (Integer num : numbers) { if (num % 2 == 0) { result.add(num); } } return result; }

5. Consider the following pseudo code for a temperature control system: temperature = readTemperature() desiredTemp = 22 // TODO: Add conditional logic here What conditional logic would be most appropriate to add in order to adjust the heating or cooling based on the current temperature?

   A.if (temperature < desiredTemp) { turnOnHeating() } else { turnOnCooling() }

B. while (temperature != desiredTemp) { adjustTemperature() }

C.if (temperature < desiredTemp) { turnOnHeating() } else if (temperature > desiredTemp) { turnOnCooling() } else { maintainTemperature() }

D.switch(temperature) { case < desiredTemp: turnOnHeating(); break; case > desiredTemp: turnOnCooling(); break; default: maintainTemperature(); }

6. You're developing a Java method to categorize products based on their price. The categories are: 'Budget' for prices under $50, 'Mid-range' for prices between $50 and $200, and 'Premium' for prices over $200. Which of the following implementations is the most efficient and readable?

A.public String categorizeProduct(double price) { if (price < 50) return "Budget"; if (price >= 50 && price <= 200) return "Mid-range"; if (price > 200) return "Premium"; return "Unknown"; }

B.public String categorizeProduct(double price) { String category; switch ((int) price) { case 0: case 1: case 2: case 3: case 4: category = "Budget"; break; case 5: case 6: case 7: case 8: case 9: case 10: category = "Mid-range"; break; default: category = "Premium"; } return category; }

C.public String categorizeProduct(double price) { if (price < 50) { return "Budget"; } else if (price <= 200) { return "Mid-range"; } else { return "Premium"; } }

D.public String categorizeProduct(double price) { while (true) { if (price < 50) return "Budget"; if (price >= 50 && price <= 200) return "Mid-range"; if (price > 200) return "Premium"; } }

7. You are developing a method to process customer orders in an e-commerce application. The method needs to apply different discounts based on the total order amount and the customer's membership status. The discount rules are as follows: - For orders over $100, apply a 5% discount - For orders over $200, apply a 10% discount - For premium members, add an additional 5% discount on top of any other discount Which of the following code snippets correctly implements these discount rules?

A.if (orderAmount > 200) discount = 0.10; else if (orderAmount > 100) discount

= 0.05; if (isPremiumMember) discount += 0.05;

B.if (orderAmount > 100) discount = 0.05; if (orderAmount > 200) discount = 0.10; if (isPremiumMember) discount = discount + 0.05;

C.discount = (orderAmount > 200) ? 0.10 : (orderAmount > 100) ? 0.05 : 0; if (isPremiumMember) discount += 0.05;

D.discount = 0; if (orderAmount > 100) discount = 0.05; if (orderAmount > 200) discount = 0.10; if (isPremiumMember) discount *= 1.05;

8. You are developing a weather application that needs to provide clothing recommendations based on temperature. The recommendations should be as follows: - Below 0°C: "Heavy winter coat" - 0°C to 10°C: "Light jacket" - 11°C to 20°C: "Sweater" - Above 20°C: "T-shirt" Which of the following code snippets correctly implements this logic?

A.if (temp < 0) print("Heavy winter coat"); else if (temp < 10) print("Light jacket"); else if (temp < 20) print("Sweater"); else print("T-shirt");

B.switch (temp) { case < 0: print("Heavy winter coat"); break; case < 10: print("Light jacket"); break; case < 20: print("Sweater"); break; default: print("T-shirt"); }

C.when { temp < 0 -> print("Heavy winter coat") temp in 0..10 -> print("Light jacket") temp in 11..20 -> print("Sweater") else -> print("T-shirt") }

D.if (temp <= 0) print("Heavy winter coat"); else if (temp <= 10) print("Light jacket"); else if (temp <= 20) print("Sweater"); else print("T-shirt");