

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Scheme of Valuation/Answer Key**  (Scheme of evaluation (marks in brackets) and answers of problems/key) | | | | | |
| **APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY**  FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2019 | | | | | |
| **Course Code: CS206** | | | | | |
| **Course Name: OBJECT ORIENTED DESIGN AND PROGRAMMING (CS)** | | | | | |
| Max. Marks: 100 | | |  | Duration: 3 Hours | |
| **PART A** | | | | | |
|  |  | ***Answer all questions, each carries 3 marks.*** | | | Marks |
| 1 |  | Explain polymorphism(1.5) – Explain Encapsulation(1.5) | | | (3) |
| 2 |  | Any three relevant classes represented using rectangles (1.5) + Relationship between classes (1.5). | | | (3) |
| 3 |  | Call by value(1.5) + Example(1.5) | | | (3) |
| 4 |  | Parameterized constructor - Definition (1)      Parameterized constructor - Example (1)      Yes, it is possible to define a parameterized constructor for a class without defining a parameter-less constructor (1) | | | (3) |
|  |  |  | | |  |
| **PART B** | | | | | |
| ***Answer any two full questions, each carries 9 marks.*** | | | | | |
| 5 | a) | Representing the system boundary using rectangle (1) + Any 2 actors (2) + Any 6 use cases (3). | | | (6) |
|  | b) | Java Virtual Machine(3) | | | (3) |
| 6 | a) | Life Cycle(4) | | | (4) |
|  | b) | Program (5) | | | (5) |
| 7 | a) | Method overloading - Definition (1) + Example (2) | | | (3) |
|  | b) | Default constructor(1) + parameterized constructor(1) + en-queue(1) + de-queue(1)+ display(1) : Object creation and calling the function(1) | | | (6) |
| **PART C** | | | | | |
| ***Answer all questions, each carries 3 marks.*** | | | | | |
| 8 |  | Packages(1.5) + Creation of packages(1.5) | | | (3) |
| 9 |  | Interfaces(1.5) + Usage of interfaces(1.5) | | | (3) |
| 10 |  | Checked exceptions are those exceptions that must be included in a method’s throws list if that method can generate one of these exceptions and does not handle it itself (2) + Example(1) | | | (3) |
| 11 |  | Explanation of Thread Priorities - Any two valid points (2) + Assignment of thread priorities is done using setPriority( ) (1) | | | (3) |
| **PART D** | | | | | |
| **Answer any two full questions, each carries 9 marks.** | | | | | |
| 12 | a) | Explanation of default(1) + private(1) + public(1) + protected(1) | | | (4) |
|  | b) | Use of extends(2) + Usage of super for calling constructor of superclass (1.5) usage of super to call overridden methods and hidden variables(1.5) | | | (5) |
| 13 | a) | Any two stream classes along with one or two methods(2 \*2) | | | (4) |
|  | b) | Odd thread (2.5) + Even thread (2.5)  30% marks may be granted for the student's effort to create a thread in Java using run( ) method, start( ) method etc. | | | (5) |
| 14 | a) | Abstract classes(3) | | | (3) |
|  | b) | New exception creation (2) + Correct Program(4) | | | (6) |
| **PART E** | | | | | |
| **Answer any four full questions, each carries 10 marks.** | | | | | |
| 15 | a) | Source generates an event  (1)  + When an event occurs, all registered listeners are notified by source (1) + Event (1) + Listener must have been registered with one or more sources (1) + Listener must implement abstract methods in necessary listener interfaces (1) + Example of event and source (1). | | | (6) |
|  | b) | Four methods(4\*1) | | | (4) |
| 16 | a) | <applet > tag with <param> (2.5) Use of getParameter() (2.5) | | | (5) |
|  | b) | Creation of Font by creating object of the class Font (2) + Any 3 methods in the class Font or any 3 valid points about Font class (3). | | | (5) |
| 17 | a) | Layout manager(2) + Explanation with suitable code(3) | | | (5) |
|  | b) | Adapter classes(2) + WindowAdapter class(3) | | | (5) |
| 18 | a) | Dynamic queries(1) + PreparedStatement and usage(4) | | | (5) |
|  | b) | Applet life cycle - diagram (1) + need for init() method (1) + need for start() method (1) +  need for destroy() method (1) +  need for stop() & paint() method (1). | | | (5) |
| 19 | a) | Steps (4) + Sample Code Segment (3) | | | (7) |
|  | b) | Difference(3) | | | (3) |
| 20 | a) | Paint() -(2) + repaint(1) | | | (3) |
|  | b) | Any two AWT controls - (Name of an AWT control (0.5) + name of any one event generated by an AWT control (0.5) + Demonstration of how any one event generated by an AWT control is handled with suitable code fragment (2.5)) x 2 | | | (7) |
| \*\*\*\* | | | | | |