2. (15 points) Below is the UML diagram for a class. Write a class header for the described class. Assume all of the required data types have already been defined. There is no need to write accessor methods. ParticleEmitter LocY: int LocY: int Angle: double Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle -ExamineParticle(): void	1.	(15 Points) We have discussed both GDI with each.	B and Valgrind in class. Briefly describe which sorts of bugs you'd find
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
ParticleEmitter -LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle			
-LocX: int -LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle	2.		
-LocY: int -Angle: double -Type: Particle +ParticleEmitter() +ParticleEmitter(x: int, y: int) +Emit(): Particle		ParticleEmitter	
+ParticleEmitter(x : int, y : int) +Emit() : Particle		-LocY : int -Angle : double	
		+ParticleEmitter(x : int, y : int) +Emit() : Particle	