**Locker Management System**

1. **Laptop**
   * Attributes:
     + serialNumber: String
     + currentBattery: int (mAh)
     + maxBattery: int (mAh)
   * Methods:
     + Laptop(serialNumber: String, currentBattery: int, maxBattery: int)
     + getBatteryPercentage(): double
2. **Locker**
   * Attributes:
     + lockerNumber: int
     + laptop: Laptop (optional)
   * Methods:
     + Locker(lockerNumber: int)
     + storeLaptop(laptop: Laptop): boolean
     + removeLaptop(): Laptop
3. **LockerBlock**
   * Attributes:
     + lockers: List<Locker>
     + chargerCapacity: int = 522 mAh
   * Methods:
     + LockerBlock()
     + addLocker(locker: Locker): void
     + storeLaptopInLocker(laptop: Laptop, lockerNumber: int): boolean
     + getAvailableLockers(): List<Locker>
     + getFirstAvailableLocker(): Locker
     + calculateChargingTime(): double

**Gaming PC Representation System**

**1. Motherboard**

* **Attributes**:
  + maxSlots: int
  + usedSlots: int
  + graphicsCards: List<GraphicsCard>
  + ssds: List<SSD>
  + soundCards: List<SoundCard>
* **Methods**:
  + Motherboard(maxSlots: int)
  + addGraphicsCard(graphicsCard: GraphicsCard): void
  + addSSD(ssd: SSD): void
  + addSoundCard(soundCard: SoundCard): void
  + getAvailableSlots(): int

**2. GraphicsCard**

* **Attributes**:
  + maxGHz: double
  + videoMemory: int (MB)
* **Methods**:
  + GraphicsCard(maxGHz: double, videoMemory: int)

**3. SSD**

* **Attributes**:
  + capacity: int (MB)
  + maxCapacity: int (MB)
  + usedSpace: int (MB)
* **Methods**:
  + SSD(capacity: int, maxCapacity: int)
  + getRemainingSpace(): int

**4. SoundCard**

* **Attributes**:
  + channels: int
* **Methods**:
  + SoundCard(channels: int)

**5. Game**

* **Attributes**:
  + title: String
  + requiredSpace: int (MB)
  + requiredVideoMemory: int (MB)
* **Methods**:
  + Game(title: String, requiredSpace: int, requiredVideoMemory: int)

**6. GamingPC**

* **Attributes**:
  + motherboard: Motherboard
  + installedGames: List<Game>
* **Methods**:
  + GamingPC(motherboard: Motherboard)
  + installGame(game: Game): void
  + hasEnoughSpace(): boolean