BITMASKRPG – A JAVA-BASED EDUCATIONAL RPG FEATURING BITWISE OPERATIONS AND TURN-BASED COMBAT

Brabete.Ma.Alexandru@student.utcluj.ro

TEHNICAL UNIVERSITY OF CLUJ- NAPOCA, DEPARTMENT OF SPECIALIZATION WITH PSYCHOPEDAGOGICAL PROFILE

Table of Contents

Introduction

BitMaskRPG is a creative blend of classic programming concepts and game design, developed entirely in Java. It uses bitwise operations to drive character abilities and simulate a graphical turn-based combat system, offering both educational value and interactive fun.

This application is especially suited for learning binary logic in an engaging way, while also introducing core game development principles.

Objectives

- Help users understand bitwise operations through visual and interactive gameplay.
- Build a modular RPG game using clean object-oriented principles.
- Provide a replayable, GUI-based battle simulator with animated health tracking and power-ups.
- Create an extensible base for adding more RPG elements like skills, spells, and multiplayer.

Technologies Used

Technology	**Description**
Java	Core programming language
Java Swing	GUI framework for building desktop UI
Timer (javax.swing)	Handles combat turn intervals
Collections	Inventory management (ArrayList)
Object-Oriented Design	Modular and maintainable code structure

Key Features

- Input any two integers for player initialization
- Compute and display bitwise operations (AND, OR, XOR, NOT)
- Turn-based animated combat simulation
- Inventory system (Healing Potions, Armor)

- Health and armor tracking with JProgressBar
- Output log area to show real-time combat events
- Reset system to start fresh without restarting the program
- Randomized reward system to increase unpredictability and engagement

Architecture & Design

The application is structured with a single main class: BitMaskRPG. This class is divided into multiple well-defined methods:

- main(): Initializes the GUI and sets up event listeners.
- handleBitwiseOperations(): Parses inputs and performs bitwise operations.
- startBattle(): Starts a timed loop for turn-based combat.
- calculateDamage(): Logic to determine how much HP is reduced per attack.
- resetGame(): Resets the entire game state (except class reload).
- updateDialogue(String): Updates the event log area with new messages.
- addItemToInventory(String, int): Adds items to player inventories based on random drop logic.

Game Flow

1. Start App
   Players see input fields and buttons in the GUI.
2. Enter Numbers
   Each player inputs a number, which the game interprets using bitwise operations.
3. Apply Bitwise Logic
   The output area displays results of bitwise AND, OR, XOR, NOT.
4. Inventory Drop
   A random item (e.g., Armor, Healing Potion) is added to a player's inventory.
5. Start Battle
   Players take turns attacking. Stats are shown in the GUI with animated health updates.
6. Battle Outcome
   When one player reaches 0 HP, the other is declared the winner and earns a stat boost.

7. Reset Option
   Game can be reset at any time to replay with new values and fresh inventory.

Bitwise Operations Explained

Operation	Description	Java Code Example
AND	Both bits must be 1	num1 & num2
OR	At least one bit must be 1	`num1
XOR	Exactly one bit must be 1	num1 ^ num2
NOT	Flips all bits (1 becomes 0 and vice versa)	~num1

Combat System Mechanics

- Each player has:
  - Health: Starts at 100
  - Attack Power: Base 20
  - Defense & Armor: Gained through inventory drops or victories
- Combat loop:
  - Every second, both players attack
  - Damage = attackPower - defense (minimum damage = 1)
  - Health bars (JProgressBar) update in real-time

Inventory & Progression

Inventory uses ArrayList<String> to manage items like:

- Healing Potion: Can be used later to restore HP (future functionality)
- Armor: Automatically improves player defense or is added to stats after battle

Future upgrades may allow:

- Item crafting

- Equipment upgrades
- Consumable buffs

Graphical User Interface (GUI)

Elements used:

- JFrame – main window
- JTextField – for user input
- JTextArea – to show battle logs and bitwise output
- JButton – for actions: bitwise operations, start battle, reset
- JProgressBar – real-time health tracking

Skills Gained

BitMaskRPG – Java Desktop RPG with Bitwise Logic
Technologies: Java, Swing, Bitwise Math, Event-Driven Programming, OOP

- Developed a complete GUI RPG game using Java Swing and bitwise arithmetic.
- Applied binary logic and low-level operations for character mechanics.
- Designed a turn-based battle system with randomized loot and animated HP bars.
- Practiced real-time UI updates using event listeners and javax.swing.Timer.
- Implemented modular game architecture using object-oriented best practices.
- Strengthened Java skills in user input validation, GUI layout design, and game logic modeling.

Future Enhancements

- Add support for:
    - Magic attacks or elemental powers
    - Multiplayer networking
    - Save/load inventory between sessions using serialization
    - Custom character creation

- Sound effects and battle animations
- Experience points and leveling system

Conclusion

BitMaskRPG showcases the fusion of educational programming concepts with gamification. It is a demonstration of how low-level binary logic can be applied creatively within high-level game mechanics, making it both a learning tool and a fun project.