Problem Statement 1:

Design a pseudocode algorithm for a simple inventory management system. The system should allow users to add new items, update quantities, and generate reports. Implement functions for each operation, and incorporate error handling using exceptions.

Psudocode:

```
START
  DECLARE inventory as a map with keys as itemNames and values as quantities
  DECLARE loopOn as BOOLEAN
  SET loopOn = TRUE

  FUNCTION AddItem(STRING itemName, INTEGER quantity):
    IF itemName exists in inventory:
      PRINT “Item already exists. Use UpdateQuantity to modify it.”
    ELSE:
      SET inventory[itemName] = quantity
      PRINT “Item added successfully.”

  FUNCTION UpdateQuantity(STRING itemName, INTEGER quantity):
    IF itemName exists in inventory:
      SET inventory[itemName] = quantity
      PRINT “Quantity updated successfully.”
    ELSE:
      PRINT “Error: Item not found.”

  FUNCTION GenerateReport():
    PRINT “Inventory Report:”
```

```
    IF inventory is empty:
        PRINT “Inventory is empty.”
    ELSE:
        FOR each itemName in inventory:
            PRINT itemName + “: ” + inventory[itemName]

FUNCTION Main():
    WHILE loopOn:
        PRINT “1. Add Item”
        PRINT “2. Update Quantity”
        PRINT “3. Generate Report”
        PRINT “4. Exit”
        INPUT choice

        SWITCH choice:
            CASE 1:
                INPUT itemName
                INPUT quantity
                CALL AddItem(itemName, quantity)
                BREAK
            CASE 2:
                INPUT itemName
                INPUT quantity
                CALL UpdateQuantity(itemName, quantity)
                BREAK
            CASE 3:
                CALL GenerateReport()
                BREAK
            CASE 4:
```

```
            SET loopOn = FALSE
            PRINT “Exiting system.”
            BREAK
        DEFAULT:
            PRINT “Invalid choice. Try again.”

    CALL Main()
END
```