## **Problem 1: Secure Bank Account**

**Problem Statement:**  
You are required to implement a BankAccount class that securely manages an account balance.

**Class Specifications:**

* + Stores the account number.
  + Stores the balance amount.
  + Initializes the account number and balance.
  + Adds the given amount to the balance.
  + Withdraws the given amount if sufficient balance exists; otherwise, return "Insufficient funds".
  + Returns the current balance.

**Input Format:**

* First, a string representing the account number.
* Second, a float representing the initial balance.
* Then, multiple queries in the format:
  + "D amount" (Deposit)
  + "W amount" (Withdraw)
  + "B" (Get Balance)

**Output Format:**

* Print the balance after each "B" query.
* Print "Insufficient funds" if a withdrawal fails.

**Example:**

Input:

123456

1000

D 500

W 200

B

W 2000

B

Output:

1300

Insufficient funds

1300

**Problem 3: Employee Management System**

**Problem Statement:**  
You need to implement an Employee class that securely manages employee details.

**Class Specifications:**

* **Stores the employee ID** as a string.
* **Stores the employee salary** as a float.
* **Initializes the employee ID and salary.**
* **Increases salary** by a given percentage.
* **Gets the current salary.**

**Input Format:**

* First, a string representing the employee ID.
* Second, a float representing the initial salary.
* Then, multiple queries in the format:
  + "I percentage" → Increase salary by given percentage.
  + "S" → Get the current salary.

**Output Format:**

* Print the updated salary after each "S" query.

**Example:**

Input:

EMP001

5000

I 10

S

I 5

S

Output:

5500.0

5775.0

**Problem 6: Smart Home System**

**Problem Statement:**  
You are required to implement a SmartHome system that securely manages different smart devices in a home.

**Class Specifications:**

* **Stores the device name.**
* **Toggles the light between ON and OFF.**
* **Stores the temperature and allows updates to it.**
* **Stores a list of devices and allows adding new devices.**
* **Controls a device (toggle lights or set thermostat temperature).**
* **Lists all devices.**

**Input Format:**

* "Light name" → Create a light device.
* "Thermostat name" → Create a thermostat device.
* "Add name" → Add a device to the smart home.
* "List" → List all added devices.
* "Toggle name" → Toggle a light ON/OFF.
* "SetTemp name value" → Set temperature for thermostat.

**Output Format:**

* Print responses based on the command.

**Example:**

Input:

Light LivingRoomLight

Thermostat HomeThermostat

Add LivingRoomLight

Add HomeThermostat

List

Toggle LivingRoomLight

SetTemp HomeThermostat 25

Toggle LivingRoomLight

Output:

Devices: LivingRoomLight, HomeThermostat

Light LivingRoomLight is now ON

Thermostat HomeThermostat is set to 25°C

Light LivingRoomLight is now OFF