**Đề bài: Quản lý Tài khoản Ngân hàng**

**Mô tả:**  
Bạn được yêu cầu viết chương trình quản lý một tài khoản ngân hàng với các yêu cầu sau:

* Tài khoản có số dư (balance) ban đầu.
* Cần bảo vệ số dư, không cho truy cập trực tiếp từ bên ngoài (tức là sử dụng tính đóng gói – encapsulation).
* Hỗ trợ hai loại giao dịch: nạp tiền (deposit) và rút tiền (withdraw).
* Khi rút tiền, nếu số dư không đủ, thông báo "Insufficient funds".

**Yêu cầu:**

1. Viết một lớp Account với thuộc tính private \_\_balance.
2. Lớp Account phải có hai phương thức:
   * deposit(amount): nạp tiền vào tài khoản và in ra số tiền vừa nạp cùng số dư mới.
   * withdraw(amount): rút tiền từ tài khoản nếu đủ số dư, in ra số tiền vừa rút và số dư còn lại; nếu không đủ, in ra "Insufficient funds".

**Cú pháp in ra:**

* Sau khi nạp tiền: Deposited: <số tiền>, New Balance: <số dư mới>
* Sau khi rút tiền thành công: Withdrew: <số tiền>, Remaining Balance: <số dư còn lại>
* Khi rút tiền không thành công: Insufficient funds

**Input:**

* Dòng đầu tiên là số nguyên B (0 ≤ B ≤ 10^9) – số dư ban đầu của tài khoản.
* Dòng thứ hai là số nguyên N (1 ≤ N ≤ 1000) – số lượng giao dịch.
* N dòng tiếp theo, mỗi dòng gồm một ký tự D hoặc W (tương ứng deposit hoặc withdraw) và một số nguyên X (1 ≤ X ≤ 10^9) – số tiền thực hiện giao dịch.

**Output:**

* In ra kết quả của từng giao dịch theo cú pháp đã cho.

**Ví dụ Input:**

100

3

D 50

W 100

W 100

**Ví dụ Output:**

Deposited: 50, New Balance: 150

Withdrew: 100, Remaining Balance: 50

Insufficient funds

**Ghi chú:**

* Bạn phải đảm bảo tính đóng gói của số dư (balance) bằng cách dùng thuộc tính private.
* Các phương thức phải kiểm tra điều kiện trước khi thực hiện giao dịch.