Print the final number xyzw...

Ex:-

n = 7

int ans = 0 [ans = (ans * 10) + num]

num = 3 , ans = 0 * 10 + 3 = 3

num = 2 , ans = 3 * 10 + 2 = 32

num = 0 , ans = 32 * 10 + 0 = 320

num = 5 , ans = 320 * 10 + 5 = 3205

num = 6 , ans = 3205 * 10 + 6 = 32056

num = 7 , ans = 32056 * 10 + 7 = 320567

num = 7 ans = 320567 * 10 + 7 = 3205677

Code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int ans = 0;
    for (int i = 0; i < n; i++) {
        int num = scn.nextInt();
        ans = ans * 10 + num;
    }
    System.out.println(ans);
}
```

Reverse n-digit number

Imp

n = 5

int ans = 0 , (ans = ans * 10 + num)

- num = 1 , ans = 0 * 10 + 1 = 1
- num = 5 , ans = 1 * 10 + 5 = 15
- num = 7 , ans = 15 * 10 + 7 = 157
- num = 3 , ans = 157 * 10 + 3 = 1573
- num = 2 , ans = 1573 * 10 + 2 = 15732

$n = 15732 > 0$

Imp

int reverse = 0

int rem = n % 10 , reverse = reverse * 10 + rem , n = n / 10

int rem = 2 , reverse = 0 * 10 + 2 = 2 , n = 1573 > 0

int rem = 3 , reverse = 2 * 10 + 3 = 23 , n = 157 > 0

int rem = 7 , reverse = 23 * 10 + 7 = 237 , n = 15 > 0

int rem = 5 , reverse = 237 * 10 + 5 = 2375 , n = 1 > 0

int rem = 1 , reverse = 2375 * 10 + 1 = 23751 , n = 0 > 0 ✗

condition: $n > 0$

Code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();

    int ans = 0;
    for (int i = 0; i < n; i++) {
        int num = scn.nextInt();
        ans = ans * 10 + num;
    }
    System.out.println(ans);
    int result = reverse(ans);
    System.out.println(result);
}

public static int reverse(int n) {
    int reverse = 0;

    while (n > 0) {
        int rem = n % 10;
        reverse = reverse * 10 + rem;
        n = n / 10;
    }

    return reverse;
}
```

n = 123

reverse = 0

123 > 0, rem = 3

reverse = 3

12 > 0, rem = 2

reverse = 32

1 > 0, rem = 1

reverse = 321

0 > 0, ×

Write a function to check if an Armstrong number or not

Armstrong :-

int n = 153 , digits = 3

any $\Rightarrow (1)^3 + (5)^3 + (3)^3$

any $\Rightarrow 1 + 125 + 27$

ans $\Rightarrow$ 153

n = 1634 digits = 4

$\Rightarrow (1)^4 + (6)^4 + (3)^4 + (4)^4$

$\Rightarrow$ 1634

n = 153

int rem = n % 10 , int ans = 0 , n = n / 10

rem = 3 , ans = $(3)^3$, n = 15 (n > 0)

rem = 5 , ans = $(3)^3 + (5)^3$, n = 1 (n > 0)

rem = 1 , ans = $(3)^3 + (5)^3 + (1)^3$, n = 0 (n > 0) ✗

n = 153 , ans = $(3)^3 + (5)^3 + (1)^3$

```
if ( n == ans ) return true;
else return false;
```

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    for (int i = 0; i < n; i++) {
        int num = scn.nextInt();
        boolean ans = isArmstrong(num);
        System.out.println(ans);
    }
}

public static boolean isArmstrong(int num) {
    int ans = 0;
    int temp = num;
    while (num > 0) {
        int rem = num % 10;
        ans = ans + (rem * rem * rem);
        num = num / 10;
    }

    if ( ans == temp ) {
        return true;
    } else {
        return false;
    }
}
```