

# ☑ PAT 乙级题目讲解：1016 《部分A+B》

## 🔗 题目简介

本题考查的是字符串提取 + 数字构造的模拟实现问题。

给定两个正整数  $A$  和  $B$ ，以及两个数字  $D_A$  和  $D_B$ 。我们需要从  $A$  中提取出所有为  $D_A$  的数字，并按原顺序拼接成一个新整数  $P_A$ ；同理从  $B$  中提取出所有为  $D_B$  的数字，拼接成  $P_B$ ，并输出  $P_A + P_B$ 。

## 🔍 样例分析

输入样例 1：

```
3862767 6 13530293 3
```

- $A = 3862767, D_A = 6 \rightarrow$  提取出数字 6,  $P_A = 66$ 。
- $B = 13530293, D_B = 3 \rightarrow$  提取出数字 3,  $P_B = 33$ 。
- 输出  $P_A + P_B = 66 + 33 = 99$ 。

输出：

```
99
```

输入样例 2：

```
3862767 1 13530293 8
```

- $A$  中没有数字 1,  $P_A = 0$ 。
- $B$  中没有数字 8,  $P_B = 0$ 。
- 输出  $0 + 0 = 0$ 。

输出：

```
0
```

## 🔍 解题思路

## 🔗 变量说明

变量名	数据类型	含义
a	int	输入的整数 $A$
da	int	要提取的数字 $D_A$
b	int	输入的整数 $B$
db	int	要提取的数字 $D_B$
p	int	计算得出的新整数 $P_A$ 或 $P_B$

## ☑ Step 1: 读取输入数据

从输入中读取整数  $A$  和  $B$  以及它们对应的数字  $D_A$  和  $D_B$ 。

```
cin >> a >> da >> b >> db;
```

## ☑ Step 2: 定义函数 p(x, dx) 提取部分数字

定义一个函数 `p` 来提取给定整数  $x$  中所有为  $dx$  的数字，并拼接成一个新整数。

```
int p(int x, int dx) {  
    int px = 0;  
    while(x) {  
        if(x % 10 == dx) {  
            px = px * 10 + dx;  
        }  
        x /= 10;  
    }  
    return px;  
}
```

- 每次从  $x$  中提取最低位数字，如果数字等于  $dx$ ，就拼接上；
- 最后返回拼接得到的结果。

## ☑ Step 3: 计算并输出 $P_A + P_B$

通过调用 `p(a, da)` 和 `p(b, db)` 分别得到  $P_A$  和  $P_B$ ，然后输出它们的和。

```
cout << p(a, da) + p(b, db);
```

## ☑ 完整代码

```
#include <bits/stdc++.h>  
using namespace std;  
  
int a, da, b, db;
```

```
int p(int x, int dx) {
    int px = 0;
    while(x) {
        if(x % 10 == dx) {
            px = px * 10 + dx;
        }
        x /= 10;
    }
    return px;
}

int main() {
    cin >> a >> da >> b >> db;
    cout << p(a, da) + p(b, db);
}
```

## 🚩 常见错误提醒

错误类型	具体表现
字符串提取错误	提取数字时未按正确顺序拼接，导致结果错误
输出格式错误	输出时未按照题目要求格式打印，可能丢失空格或换行

## ✅ 总结归纳

### 🔑 核心方法总结

通过字符串提取模拟提取数字，使用数位拆分和重组来拼接结果。

### 📋 技术要点回顾

- 数位拆分与拼接的标准方法；
- 数字匹配与提取技巧；
- 避免溢出的边界条件处理。

### 📊 复杂度分析

- 时间复杂度：  $\mathcal{O}(n)$
- 空间复杂度：  $\mathcal{O}(1)$

其中  $n$  为输入数据中最大数字的位数（最多 9 位）。

## 🧠 思维拓展

- 扩展思考**：如果题目要求提取数字并按奇偶性筛选拼接，该如何实现？
- 类似题目**：数字构造类问题，如回文数判断、数字逆序等；
- 其他思路**：可将函数 `p` 中的数位提取改为直接使用字符串操作。