

## A+B Problem (a+b.cpp)

总时间限制: 1s

内存限制: 64MB

### 【问题描述】

输入两个整数  $a, b$ ，输出它们的和 ( $|a|, |b| \leq 10^6$ )。

### 【输入格式】

两个整数以空格分开。

### 【输出格式】

一个数，即  $a+b$  的结果。

### 【输入样例】(a+b.in)

20 30

### 【输出样例】(a+b.out)

50

范例程序（一）c 语言版: a+b.c

```
#include <stdio.h>
```

```
int main() {
    int a,b;
    freopen("a+b.in","r",stdin);           //输入重定向，文件读入
    freopen("a+b.out","w",stdout);         //输出重定向，文件写入
    scanf("%d%d",&a,&b);
    printf("%d", a+b);
    fclose(stdin);
    fclose(stdout);
    return 0;
}
```

范例程序（二）c++语言版: a+b.cpp

```
#include <cstdio>
```

```
using namespace std;
```

```
int main() {
    int a,b;
    freopen("a+b.in","r",stdin);           //输入重定向，文件读入
    freopen("a+b.out","w",stdout);         //输出重定向，文件写入
    scanf("%d%d",&a,&b);
    printf("%d", a+b);
    fclose(stdin);
    fclose(stdout);
    return 0;
}
```

### 范例程序（三）c++语言版：a+b.cpp

```
#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    int a,b;
    freopen("a+b.in","r",stdin);          //输入重定向，文件读入
    freopen("a+b.out","w",stdout);         //输出重定向，文件写入
    cin>>a>>b;
    cout<<a+b;
    fclose(stdin);
    fclose(stdout);
    return 0;
}
```

---

### 倒序整数(num.cpp)

总时间限制：1s      内存限制：64MB

#### 【问题描述】

从键盘输入一个 5 位正整数，依次从低位到高位输出这个 5 位正整数每个数字。

例如：输入 12345，输出 54321。

#### 【输入格式】

只有一行，输入一个五位正整数。（不允许分五次输入各个数位上的数字，即只能一次性输入一个 5 位数给对应变量）

#### 【输出格式】

倒序输出这个五位数。

#### 【输入样例 1】(num.in)

12345

#### 【输出样例 1】(num.out)

54321

#### 【输入样例 2】(num.in)

50000

#### 【输出样例 2】(num.out)

00005

## 加密(encryption.cpp)

总时间限制: 1s

内存限制: 64MB

### 【问题描述】

你的任务是将一串字符（长度为 5）译成密码。

密码规律：用原来的字符后面第 4 个 ASCII 码对应的字符代替原来的字符，例如，字符 ‘A’ 后面第 4 个字符是 ‘E’，用 ‘E’ 代替 ‘A’。

### 【输入格式】

只有一行，输入一个长度为 5 的单词。

### 【输出格式】

输出对应的密码。

### 【输入样例】(encryption.in)

China

### 【输出样例】(encryption.out)

Glmre

---

## 最大数(maxnum.cpp)

总时间限制: 1s

内存限制: 64MB

### 【问题描述】

编程实现输入五个整数，找出这五个数的最大数和最小数。

### 【输入格式】

只有一行，五个整数，每个数之间用空格隔开。

### 【输出格式】

输出两行，第一行是最大数，第二行是最小数（具体格式参见输出样例）。

### 【输入样例】(maxnum.in)

5 8 78 1002 -56

### 【输出样例】(maxnum.out)

max=1002

min=-56