第四章 函数

第49课 变量地址、指针及引用

《信息学奥赛一本通·编程启蒙 C++版》

一、传值调用

这种调用方式是将实参的数据值传递给形参，即将实参值拷贝一个副本存放在被调用函数的栈区中。在被调用函数中，形参值可以改变，但不影响主调函数的实参值。参数传递方向只是从实参到形参，简称单向值传递。

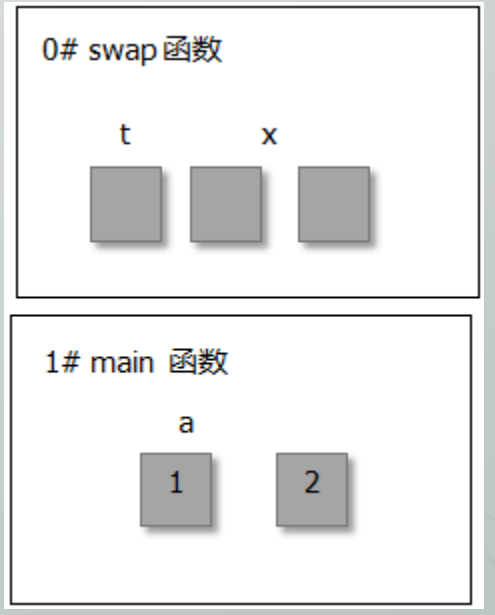
回顾第 46 课中例举的函数中使用的都是“传值调用”，下面我们分析一下，用以下函数交换两个数的程序不成功的原因，代码如下：

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. void swap(int x,int y){
4.     int t;
5.     t=x;
6.     x=y;
7.     y=t;
8. }
9. int main(){
10.    int a,b;
11.    a=1;b=2;
12.    swap(a,b);
13.    cout<<a<<' '<<b;
14.    return 0;
15.}
```

当程序从 main 函数进入，这时只有#0 main 栈帧，变量 a 和 b 在 main 栈帧里。

执行到第 12 行时，在 main 函数中的变量 a 和 b 是第 12 行函数 swap(a,b) 的实参，当程序去调用 swap 函数时，这两个变量的值会传递给第三行 swap(int x,int y)函数的形参 x 和 y,这时会产生新的栈帧,#0 栈帧成了 swap 函数的栈帧，变量 x 和 y 在#0 的栈帧里，得到的值分别是 1 和 2；而 main 函数的栈帧变成了#1，变量 a 和 b 则在#1 栈帧里面。

发生交换的过程都是在 0#栈帧里面发生的，在函数 swap（）里面发生，并不会影响 1#栈帧里面的 main()函数里的变量。因此程序最后还是输出 1 和 2，并没有发生交换。



二、传址调用

这种调用方式是将实参变量的地址值传递给形参，这时形参是指针，我们前面学到的 scanf 系列函数，就用了指针作为参赛，例如：

scanf(“%d”,&x);

即让形参的指针指向实参地址，这里不再是将实参拷贝一个副本给形参，而是让形参通过地址的方式直接控制实参，这就提供了一种可以改变实参变量的值的方法。

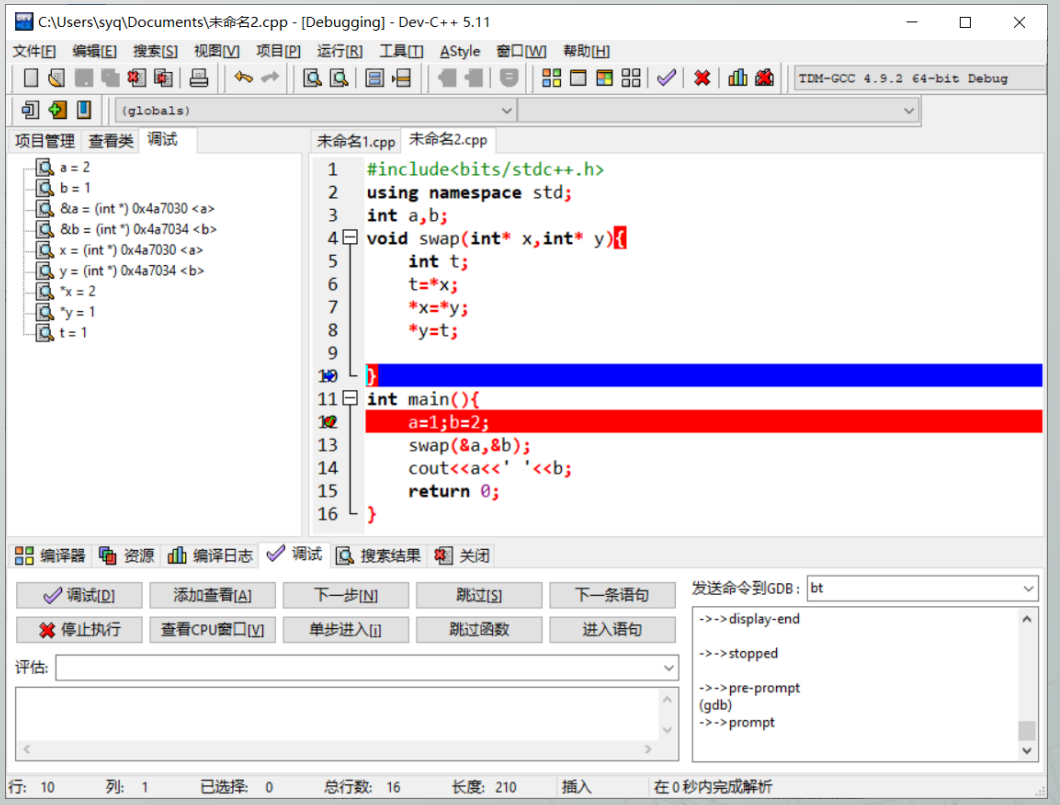
将两个数发生交换的函数改写成如下形式：

```
#include<bits/stdc++.h>
using namespace std;
int a,b;
void swap(int* x,int* y){
    int t;
    t=*x;
    *x=*y;
    *y=t;

}
int main(){
    a=1;b=2;
    swap(&a,&b);
    cout<<a<<' '<<b;
    return 0;
}
```

为了搞清楚每个变量里面具体存的值，以及他们的变化过程，我们可以借助 GDB 来查看。

先将断点设置在 12 行，并添加各个变量的查看，然后连续发送 5 次 s 命令，在单步调试的过程中，读者可以仔细观察各个变量值的变化过程。如下图所示：



三、引用

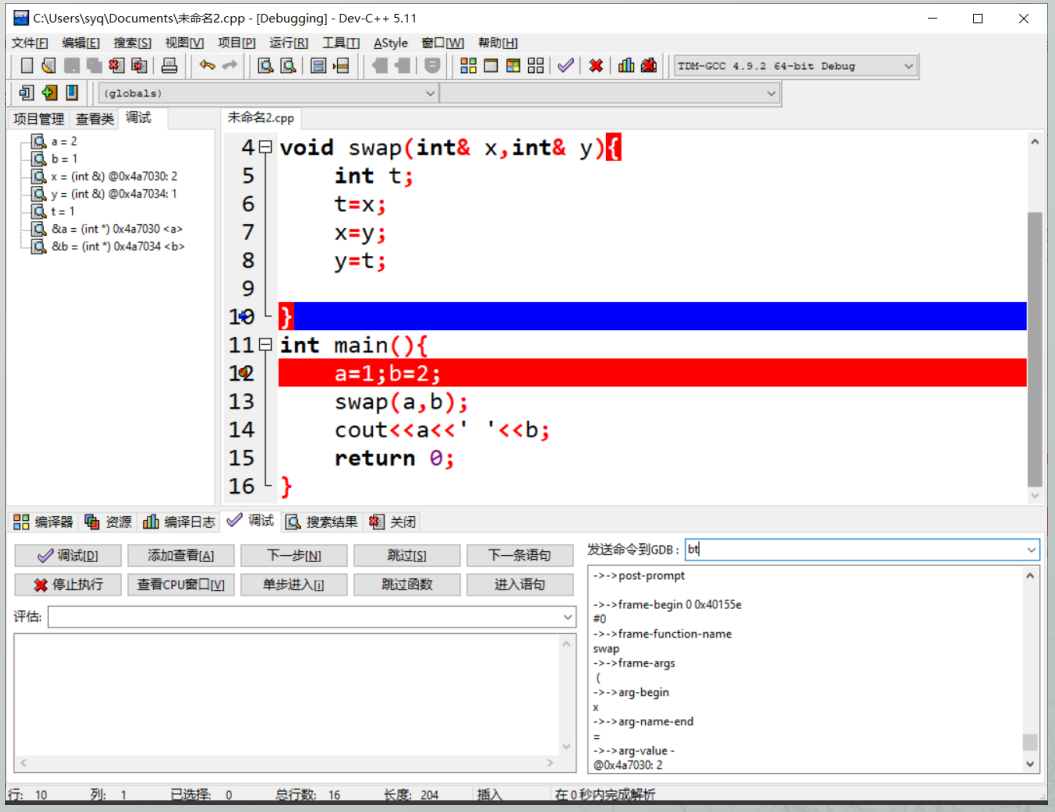
C++中的引用引用变量是一个别名，也就是说，它是某个已存在变量的另一个名字。一旦把引用初始化为某个变量，就可以使用该引用名称或变量名称来指向变量。

C++中的引用一定程度上可以代替 C 语言的指针，可以用“传引用”的方式让函数内直接修改实参。现在把两数交换的程序修改如下：

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a,b;
4. void swap(int& x,int& y){
5.     int t;
6.     t=x;
7.     x=y;
8.     y=t;
9.
10.}
11.int main(){
12.    a=1;b=2;
13.    swap(a,b);
14.    cout<<a<<' '<<b;
15.    return 0;
16.}
```

为了搞清楚每个变量里面具体存的值，以及他们的变化过程，我们依旧借助 GDB 来查看。

GDB 调试时查看到的引用和指针表示方式的异同	
引用	指针
x=(int&)@0x4a7030：2	x=(int*)0x4a7030<a>
y=(int&)@0x4a7034：1	y=(int*)0x4a7034<b>



【例 49.1】 [USACO 1.3.4]回文平方数

【题目描述】

回文数是指从左向右念和从右向左念都一样的数。如 12321 就是一个典型的回文数。

给定一个进制 B（2≤B≤20，由十进制表示），输出所有的大于等于 1 小于等于 300（十进制下）且它的平方用 B 进制表示时是回文数的数。用 A,B…… 表示 1010，1111 等。

【输入格式】

共一行，一个单独的整数 B(B 用十进制表示)

【输出格式】

每行两个 B 进制的符合要求的数字，第二个数是第一个数的平方，且第二个数是回文数。

【样例输入】

10

【样例输出】

1 1
2 4
3 9
11 121
22 484
26 676
101 10201
111 12321
121 14641
202 40804
212 44944
264 69696

【代码实现】

```
#include<bits/stdc++.h>
using namespace std;
int b,a[10000],aa[10000],cnta,cntaa;
void changeb(int x,int *a,int &cnt){
    cnt=0;
    while(x>0) a[cnt++]=x%b,x/=b;
}
bool is_pal(){
    for(int i=0,j=cntaa-1;i<j;i++,j--)
        if(aa[i]!=aa[j]) return 0;
    return 1;
}
void print(int *a,int &cnt){
    for(int i=cnt-1;i>=0;i--)
        if(a[i]>=0 && a[i]<=9) cout<<a[i];
          else cout<<char('A'+a[i]-10);
}
int main(){
    cin>>b;
    for(int i=1;i<=300;i++){
        changeb(i,a,cnta);
        changeb(i*i,aa,cntaa);
        if(is_pal()) {print(a,cnta);cout<<' ';print(aa,cntaa)
    cout<<endl;}
    }
    return 0;
}
```

谢谢！