

```
#include <stdio.h>

// We can exploit rule 6 to have functions return multiple
// values back to us. These values can be of different datatypes
void returnMulti(int *a, float *b, char *c, double *d){
    *a = 10;
    *b = 90.02;
    *c = 'F';
    *d = 123.456;
}

void foo(int a, int *ptr){
    printf("foo sees %d\n", *ptr);
    // Modifying the local copy has no effect on the original variable
    a = 100;
    // Modifying a memory location directly has affect on all clones
    // since all clones share the same memory address space
    *ptr = 300;
}

void bar(int a, int *ptr){
    printf("bar sees %d\n", *ptr);
}

void swap(int *a, int *b){
    int temp = *a;
    *a = *b;
    *b = temp;
}

int main(){
    int a = 42, b = 24;
    int *ptr = &a;

    printf("Before a = %d, b = %d\n", a, b);
    swap(&a, &b);
    printf("After a = %d, b = %d\n", a, b);

    foo(a, &a);
    bar(a, ptr);

    int p;
    float q;
    char r;
    double s;

    returnMulti(&p, &q, &r, &s);
    printf("%d\n%f\n%c\n%lf", p, q, r, s);

    return 0;
}
```