

ForNextDay 5
Stephen Cole

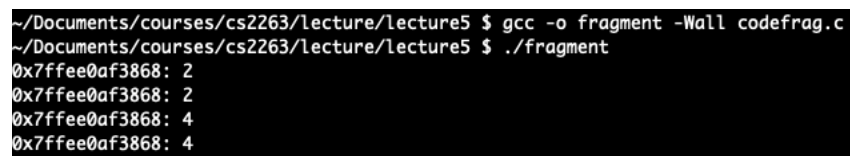
codefrag.c

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

int main(void)
{
    int i = 2;
    int* pi;
    pi = &i;
    printf("%p: %i\n", &i, i);
    printf("%p: %i\n", pi, *pi);

    *pi = 4;
    printf("%p: %i\n", &i, i);
    printf("%p: %i\n", pi, *pi);

    return EXIT_SUCCESS;
}
```



```
~/Documents/courses/cs2263/lecture/lecture5 $ gcc -o fragment -Wall codefrag.c
~/Documents/courses/cs2263/lecture/lecture5 $ ./fragment
0x7ffee0af3868: 2
0x7ffee0af3868: 2
0x7ffee0af3868: 4
0x7ffee0af3868: 4
```

Since the pointer pi is set to the same address as int i then they contain the same value. When pi is changed i is changed and vice versa.

zero.c

```
/* the value at the address in pi is
 * set to zero.
 */
void zero(int* pi){
    *pi = 0;
}

int main(void){
    int* pointer;
    zero(pointer);
    return EXIT_SUCCESS;
}
```

Stack

zero()

pi 0xFFFF

main()

pointer 0xFFFF

swap.c

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

```
#include<stdlib.h>
```

```
int main(int argc, char* argv[])
{
    int i = 10;
    int* pi = &i;
    int j = 99;
    int* pj = &j;
    printf("i = %d; j = %d\n", i, j);
    swap(pi,pj);
    printf("i = %d; j = %d\n", i, j);
    return EXIT_SUCCESS;
}
```

```
void swap(int* i, int* j)
{
    int swap = *i;
    *i = *j;
    *j = swap;
}
```

print addresses

```
int main(int argc, char* argv[])
{
    int i = 10;
    int* pi = &i;
    int j = 99;
    int* pj = &j;
    printf("i = %p; j = %p\n", pi, pj);
    swap(pi,pj);
    printf("i = %p; j = %p\n", pi, pj);
    return EXIT_SUCCESS;
}
```

myutils.c/.h

```
[scole4@gaea lecture5]$ gcc -o test -Wall test.c myutils.c
[scole4@gaea lecture5]$ ./test
2
```