

More Pointers

svn+ssh://zeus/home/CS300Public/2018/CS300PublicSVN/Pointers

myMemCpy

- Let's investigate the code
- `void myMemCpy (void *pDest, void *pSrc, int size)`

```
* ((char*) pDest) + i) = * ((char*) pSrc) + i);
```

Integer Pointers

Write a function `myIntCpy` to copy one integer to another.

What does a call look like?

String Pointers

Write a function `myStrCpy` to copy one string to another.

What does a call look like?

Pointer Questions

Consider `char name1[5], name2[] = "Jill";`

Which of the following calls will copy the string `name2` into `name1` correctly?

`myMemCpy (name1, name2, 5);`

`myMemCpy (name1, name2, strlen (name2));`

`myMemCpy (&name1, &name2, 5)`

Danger

```
char *paDoNotAlter = "ReadOnly";  
paDoNotAlter[0] = 'r';
```

Pointer Questions

Consider

```
25 char name1[5], name2[] = "Jill";  
26 char *pName3 = (char *) malloc (5);  
27  
28 myMemCpy (&name1, &name2, 5);  
29 myMemCpy (&pName3, &name2, 5);  
30  
31 printf ("%s %s\n", name1, pName3);
```

What happens when the above program segment is executed?

Pointer Questions

The following slides will use the data structures below:

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <stdlib.h>
4
5  typedef struct Person1
6  {
7      char name[5];
8      int age;
9  } Person1;
10
11 typedef struct Person2
12 {
13     char *pName;
14     int age;
15 } Person2;
16
17 void *myMemCpy (void *pDest, void *pSrc, int size)
18 {
19     int i;
20
21     for (i = 0; i < size; ++i)
22     {
23         *(char *) (pDest + i) = *(char *) (pSrc + i);
24     }
25
26     return pDest;
27 }
```


Pointer Questions

- Consider

```
25 | Person1 sPerson1 = {"Jill", 10}, sPerson2;  
26 | myMemCpy (&sPerson2, &sPerson1, sizeof (Person1));  
27 |  
28 | printf ("%s %d\n", sPerson2.name, sPerson2.age);
```

Any problems?

Pointer Questions

- Consider

```
25  Person1 sPerson1 = {"Jill", 10};  
26  Person2 sPerson2;  
27  sPerson2.pName = malloc (5);  
28  myMemCpy (&sPerson2, &sPerson1, sizeof (Person1));  
29  
30  printf ("%s %d\n", sPerson2.pName, sPerson2.age);
```

any problems?

Pointer Questions

- Consider

```
25  Person2 sPerson1, sPerson2;  
26  sPerson1.pName = malloc (5);  
27  myMemCpy (sPerson1.pName, "Jill", 5);  
28  sPerson2.age = 10;  
29  sPerson2.pName = malloc (5);  
30  myMemCpy (&sPerson2, &sPerson1, sizeof (Person1));  
31  
32  printf ("%s %d\n", sPerson2.pName, sPerson2.age);
```

any problems?