MORE POINTERS

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?

void Pointers

Write a function myMemCpy to copy any contiguous memory space 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?

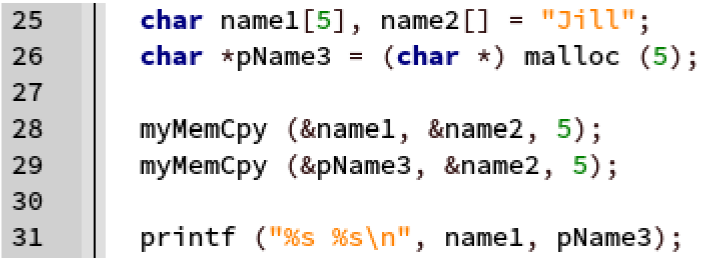
a) **myMemCpy (name1, name2, 5);**

b) **myMemCpy (name1, name2, strlen (name2));**

c) **myMemCpy (&name1, &name2, 5)**

Pointer Questions

Consider

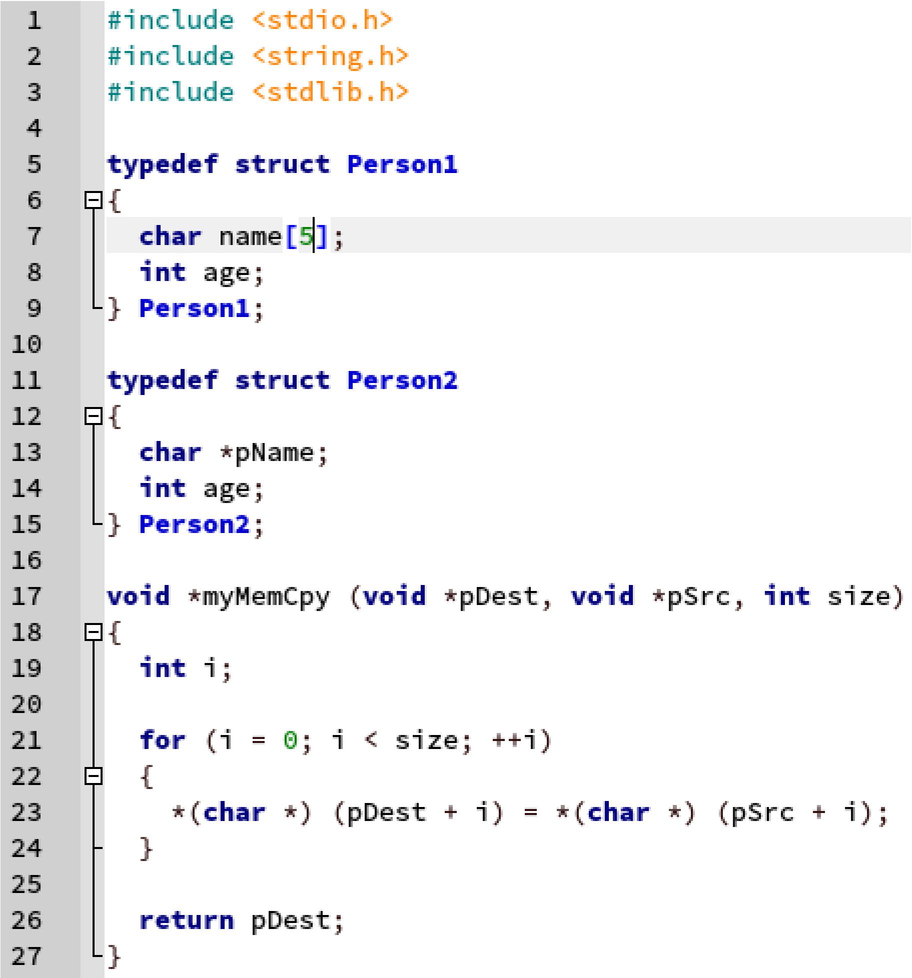


```
char name1[5], name2[] = "Jill";
char *pName3 = (char *) malloc (5);

myMemCpy (&name1, &name2, 5);
myMemCpy (&pName3, &name2, 5);

printf ("%s %s\n", name1, pName3);
```

What happens when the above program segment is executed?

Pointer Questions

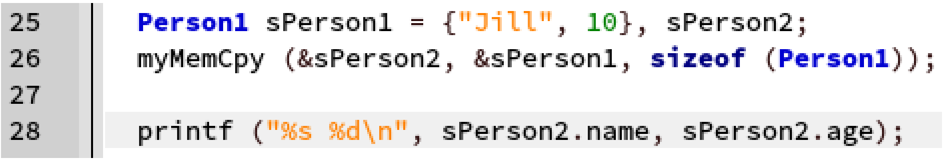
The following slides will use the data structures below:



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

typedef struct Person1
{
  char name[5];
  int age;
} Person1;

typedef struct Person2
{
  char *pName;
  int age;
} Person2;

void *myMemCpy (void *pDest, void *pSrc, int size)
{
  int i;

  for (i = 0; i < size; ++i)
  {
    *(char *) (pDest + i) = *(char *) (pSrc + i);
  }

  return pDest;
}
```

Pointer Questions

Consider

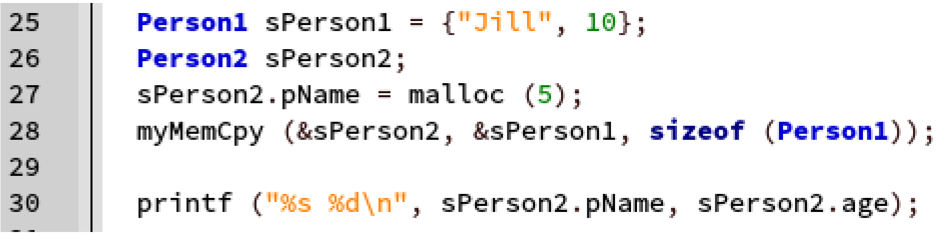


```
Person1 sPerson1 = {"Jill", 10}, sPerson2;
myMemCpy (&sPerson2, &sPerson1, sizeof (Person1));

printf ("%s %d\n", sPerson2.name, sPerson2.age);
```

Any problems?

Pointer Questions

Consider

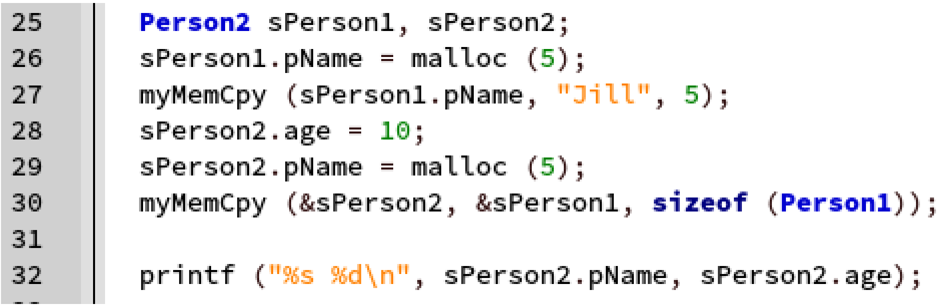


```
Person1 sPerson1 = {"Jill", 10};
Person2 sPerson2;
sPerson2.pName = malloc (5);
myMemCpy (&sPerson2, &sPerson1, sizeof (Person1));

printf ("%s %d\n", sPerson2.pName, sPerson2.age);
```

Any problems?

Pointer Questions

Consider



```
Person2 sPerson1, sPerson2;
sPerson1.pName = malloc (5);
myMemCpy (sPerson1.pName, "Jill", 5);
sPerson2.age = 10;
sPerson2.pName = malloc (5);
myMemCpy (&sPerson2, &sPerson1, sizeof (Person1));

printf ("%s %d\n", sPerson2.pName, sPerson2.age);
```

Any problems?