Courtney Makua

02/12/2022

CSCI 3731 A

Dr Edward Pier

1. Describe the difference between how Java and C++ handle arrays.

Java arrays are objects. The object holds the items in the array and the size of the array. The object also checks if the index of an accessed element is valid. C++ does not do any of this. In C++, an array is a chunck of allocated memory that doesn't keep track of the size nor does it check if an array index is valid.

2. Suppose you have a variable declared like this:
   int* n;
   Can you pass n to a function declared like this:
   void f(const int* b);
   Why or why not?

Yes, you can pass an int pointer to a const int pointer but not the other way around because if it were the other way around then the function would change the pointer which is not allowed because it is a const.

5. Suppose you have a function that takes an array as an argument in the usual way, which is declared as:

   void f(int* array, int length);

   Suppose you also have an array declared as:

   int a[128];

   How would you pass a sub-array containing only the third through sixth elements of a to the function f?

F(array+2,5);

6. What is wrong with the following code, and how would you fix it?

```
char copy[8];
const char* string = "hello";
for(int i=0; string[i] != '\0'; ++i) {
copy[i] = string[i];
}
printf("%s\n", copy);
```

copy is declared with 8 elements when it only should use 6. Change 8 to 6 and create a null terminator for string at the end.

8. What is wrong with the following code and how would you fix it?

```
double* allocateArray(int length) {
double array[length];
return array;
}
```

Only the length is passed to this function. Returning array wont do anything