**What is the reason that buffer is often defined as void\* ?**

The compiler has no idea what this pointing to

This is a general purpose point (point to a raw address)

If students answer like red colored sentence, please give full points. If he/she answer like blue colored sentence, please give 2 points

**You defined two buffers in your code**

**buf1[8] = {‘a’,’a’,’a’,’a’,’a’,’a’,’a’,’a’};**

**buf2[4] = {‘b’,’b’,’b’,’b’}**

**Now, write a code to copy buf2 to buf1 and make buf1 as {‘a’,’a’,’b’,’b’,’b’,’b’,’a’,’a’} using memcpy function.**

memcpy(&buf1[2], buf2,4)

**What are file descriptors?**

A file descriptor is an index assigned by the kernel into a table of file information maintained in the OS

File descriptors are used to reference a file when call the I/O system calls

The kernel accesses the file for the process using the file descriptor

If student’s answer is similar to any of the above colored sentence, then give full points

**Where is the main difference between the command next(n) and step(s) in gdb**

next: goes to the next instruction but doesn't dive into functions

step: goes to next instruction while diving into the function.

**You have the following code in your C program**

**char \*x = “hello\n”;**

**char x1[] = “hello\n”;**

**Which segments are the values of x and x1 stored in memory, respectively?**

\*x stores in read-only-memory

X[] stores in stack

Two points for each colored sentence

**Given an integer A=0x44113322, write its four bytes in the correct order in the memory according to the big endian system, assuming the memory address increases from the left to the right.**

Big endian system - 44113322

**Given the decimal number 30, whose binary representation is (00011110), in an 8-bit system, what is the binary representation of -30 according to two’s complement number system? Show your work.**

Ans:

First compute the complement of 00011110 -> 11100001 (3 pts)

Then add one: 1d1100010 (3 pts)

**Convert the decimal number 300 into hexadecimal, and then directly convert the hexadecimal representation into a binary representation. Show your work.**

Ans:

The algorithm for reference

Graphical user interface, text, application

Description automatically generated

(3pts) 300 = 12 16\*12 -> 0x12C

Text

Description automatically generated

(3pts) 0x12C -> 0001 0010 1100

(students need to use the above idea. If they directly convert the decimal into binary and get the correct answer, give 1pt only)

Decimal to hexadecimal:

300/16 = 18 --> Quotient, 12 --> Remainder

18/16 = 1 --> Quotient, 2 --> Remainder

1/16 = 0 --> Quotient, 1 --> Remainder

12 = c

Therefore,

12C (base 16) --> Answer

Hexadecimal to binary:

1 = 0001

2 = 0010

C = 1100

Therefore,

12C = 0001 0010 1100

**One difference between static library and dynamic library is that, when you call a library function in your code, the function in your object code is resolved at \_\_\_ time and at \_\_\_ time, respectively. To build a static library and a dynamic library, you use the command with name \_\_\_\_\_and \_\_\_\_\_\_\_, respectively, and the resulting library has the file extension \_\_\_\_ and \_\_\_\_, respectively. (one word for each blank).**

(1). link (or linking, or compilation) (2). run (or load)

(3). ar (4). gcc

(5). .a (or a) (6). .so (or so)

1.2 pt for each space.

**Assume that a file has the following access policy**

**rwxrw---x**

**Explain the above access policy for the file.**

Owner has read, write, and execute permission

Group has read and write permission

World has only execute permission

We see there are access policies for three categories. Please reduce two points for the wrong answer for each category

**Explain blocking I/O, Non-blocking I/O and Asynchronous I/O**

Blocking: The call waits for the read or write to complete before returning

Non-blocking: The call does not wait for the read or write to complete before returning

Asynchronous: I/O request returns immediately. A callback is generated when I/O completes

3 points for first colored sentence and two points for last colored sentence

The following questions are all concerning Cache.

**(a) Name the two types of cache locality and briefly explain the intuitions behind them (one sentence for each).**

* + Spatial locality: data to be accessed tend to be close to data you already accessed
  + Temporal (or time) locality: data that is accessed is likely to be accessed again soon

**(b) If the cache adopts the LFU policy, what does it mean?**

This is a cache replacement policy (2pts). It means the least frequently accessed item (or cache line) in the cache will be evicted when a new block is brought into a full cache.

If students say “This is a cache replacement policy” (2pts).

Otherwise, if the following sentence is correct, they can still get the perfect score.

**(c) Assume a memory access to main memory on a cache "miss" takes 100 ns and a memory access to the cache on a cache "hit" takes 5 ns. If 80% of the processor's memory requests result in a cache "hit", what is the average memory access time?**

5+(1-80%) \* 100 = 25ns