

粤嵌第二阶段测试题

姓名：

班级：

一、请分析以下代码：（9 分）

```
Void *func1(void *arg){
    Int a = 10;
    Pthread_join((void *)a)
}
Void *func2(void *arg){
    Char p[] = "hello";
    Pthread_exit((void *)p);
}
Void *func3(void *arg){
    Char *p = malloc(10);    Strcpy(p, "hello");
    Pthread_exit((void *)p);
}

Void main()
{
    Pthread_t t1,t2,t3;
    Pthread_create(&t1, NULL, func1, NULL);Pthread_create(&t2, NULL, func2, NULL);
    Pthread_create(&t3, NULL, func3, NULL);

    char *p;
    Pthread_join(t3, (void *)&p);
    Printf("%s\n", p);
    问题 1: printf 打印的结果是什么？为什么？
    hello , 因为线程函数中 p 分配的是堆上的空间，线程退出后返回堆上的地址

    Pthread_join(t2, (void *)&p);
    Printf("%s\n", p);
    问题 2: printf 打印的结果是什么？为什么？
    未知，因为线程函数中的 p 在进程的栈上，当线程函数退出以后，栈被回收

    Int *a;
    Pthread_join(t1, (void *)&a);
    Printf("%d\n", *a);
    问题 3: printf 打印的结果是什么？为什么？
    未知，因为线程函数中的 a 在进程的栈上，当线程函数退出以后，栈被回收

}
```

二、请简述 static 关键字的作用（3 分）

- 1) 修饰局部变量，改变局部变量的生存周期
- 2) 修饰全局变量和函数，改变作用域（作用域由原来的整个工程变为在本文件内）

三、请简要描述#include "" 和 #include <>的区别（3 分）

- 1) include "" 默认在工程目录下查找头文件，如果没有找到会去系统目录下查找头文件
- 2) Include <> 默认在系统目录下查找头文件

四、请比较动态库和静态库的区别，并且写出如果将一个动态库移植、拷贝至开发板上后需要执行哪些操作，依赖于动态库的程序才能够正常运行？（5 分）

- 1) 动态库在程序编译的时候不会被链接到目标文件中，而是在程序运行时才被载入，目标文件小
 - 2) 静态库在程序编译的时候就被链接到目标文件中，目标文件大占用空间
 - 3) 动态库的更新比静态库方便，当库文件更新时，动态库只需在目标板上更新库文件即可，程序不需要重新编译，而当静态库更新时，所有需要链接该静态库的程序需要重新编译。
- 将动态库拷贝至开发板上后需要设置环境变量

export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:动态库所在的路径

五、请比较标准 IO 和系统 IO 的区别（5 分）

- 1) 标准 IO 是标准 C 库提供的一套对文件操作的接口，带缓冲区，只要开发环境中标准 C 库，即可使用标准 IO，可移植性强。标准 IO 只能操作普通文件
- 2) 系统 IO 是由操作系统内核提供的一套操作文件的接口，不带缓冲区，各个操作系统平台提供的系统 IO 接口可能不一样，可移植性差。系统 IO 能够操作普通文件和设备文件、管道文件、socket 文件

六、请比较线程和进程的区别，且分析什么时候用多进程比较好，什么时候用多线程比较好（5 分）

- 1) 进程是系统分配资源的最小单元
 - 2) 线程是系统任务调度的最小单元
 - 3) 进程间的地址空间是互相独立的，所以进程间进行数据交换必须使用 IPC
 - 4) 线程共享进程的地址空间所以线程间数据同步不需要使用 IPC，但是需要使用互斥量、信号量等同步机制
- 至少答对以上四点
- 5) 多线程系统资源的开销要比多进程少很多
 - 6) 多线程编程时一个线程的崩溃可能导致这个进程的崩溃，但是多进程编程时不会
 - 7) 多进程占用内存多，切换复杂，cpu 利用率低，多线程反之

如果需要频繁创建和销毁的优先用多线程。需要进行大量计算的优先用多线程。

数据需要频繁进行同步优先使用线程。

任务和任务间的关联性较强优先用多线程，关联性较弱优先使用多进程

七、如果多个线程同时对一个全局变量进行自增或者自减会产生什么样的后果？为什么？我们该如何避免这种情况的发生？（线程数据的同步）（3 分）

该变量的结果不是多次自增或者自减的结果（数据没有同步），原因：全局变量是共享资源

解决方法：采用互斥量、条件变量、信号量等同步机制

八、有两台计算机的 IP 地址分别为：192.168.1.100 和 192.168.2.101，请问该采取什么样的办法能让这两台计算机互相 ping 通（2 分）

将两台计算机的子网掩码修改为：255.255.0.0

九、请分析以下代码？

```
Void *func(void *arg)
{
    int n;;
    n = *((int *)arg);
    printf("%d\n", n);}
void main()
{
    pthread_t t;
    Int num = 10;
    While (1)
    {
        //通过 socket 通信，用 num 接收客户端发送的数据
        read(sock_fd, &num, sizeof(num));
        pthread_create(&t, NULL, func, (void *)&num);
    }
}
```

请问以上代码在运行的时候有何问题？该如何改善？（注意不要考虑语法问题！）（3 分）

线程函数中 n 的值不一定是传递过来的 num 的值，因为线程函数的参数获取的是 num 的地址，而线程在运行的过程中，num 的值可能会被修改。

改进方法：将 num 定义为指针变量，每次创建线程时重新分配空间

```
Int *num ;
While (1)
{Num = (int *)malloc(sizeof(int));
read(sock_fd, num, sizeof(int));
pthread_create(&t, NULL, func, (void *)num);
}
```

十、在进行多线程编程的时候，主线程该如何回收创建的子线程的资源？如果子线程希望在退出后资源由系统自动回收该如何处理？（3 分）

1) 调用 pthread_join 函数

2) 线程分离，调用 pthread_detach()函数

十一、为什么进程间交换数据需要使用进程间通信(IPC)? 请比较进程间通信的几种方式（5 分）

1) 因为进程间的地址空间是互相独立的

2) 管道：半双工的，管道文件是一种特殊的文件，但是可以使用 write、read 等系统函数操作，编程简单。管道分为：有名管道和无名管道

消息队列：消息队列是通讯双方通过内核中维护的一个消息队列进行通信，是一种落后的通信方式

共享内存：进程间通过共享内存进行通信，数据在内核中只拷贝了一次，所以数据的交换速度非常快，但是以为共享内存是共享资源，所以需要信号量等同步机制来保证数据的同步

信号量：信号量是一种保障数据同步的机制，分为 **system V** 信号量和 **posix** 信号量

十二、什么是僵尸进程什么是孤儿进程？（2分）

1) 进程“退出”后，系统会将该进程标注为 **zombie** 状态，如果父进程调用 **wait/waitpid** 获取退出进程的状态并且回收系统资源，该进程就叫做僵尸进程

2) 父进程先于子进程退出，子进程成为了孤儿进程，孤儿进程由 **init** 进程托管

十三、父进程调用 **fork** 函数创建一个子进程以后如何防止子进程在退出以后成为僵尸进程（2分）

调用 **wait/waitpid** 函数，或者父进程通过 **signal** 注册一个 **SIGCHLD** 的信号处理函数

十四、什么是软链接，什么是硬链接，软链接和硬链接的区别是什么？（5分）

1) 软链接可以理解为快捷方式，软链接文件与被链接的文件是两个不同的文件，因为 **inode** 号不一样，软链接文件中保存的是被链接的文件的路径，如果删除被链接文件，则软链接失效

2) 硬链接文件和被链接的文件是通过一个文件，因为两个文件的 **inode** 号一样，修改其中一个文件，另外一个文件会随之修改，如果删除其中一个文件，不会影响另外一个文件，因为删除其中任意一个文件只会是该文件对应的 **inode** 节点中的硬链接数-1

十五、请写出 **TCP/IP** 模型的四层结构（2分）

应用层、传输层、网络层、数据链路层

十六、请写出 **TCP** 三次握手的过程（3分）

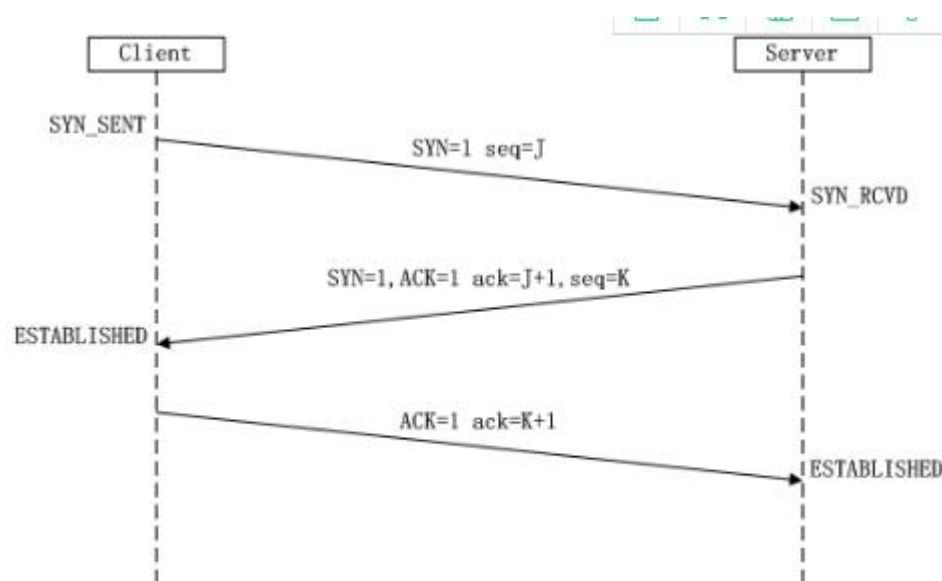


图2 TCP三次握手

十七、请比较 TCP 和 UDP 的区别 （5 分）

- 1) TCP 协议是面向连接的，而 UDP 协议是无连接的
- 2) UDP 协议是面向报文的，而 TCP 协议是面向字节流的
- 3) TCP 协议能保证传送的数据无差错、不丢失、不重复、有序到达，而 UDP 不能保证至少答对以上三点。
- 4) TCP 要求系统资源较多，UDP 较少
- 5) UDP 程序结构较简单
- 6) TCP 支持滑动窗口机制实现流量控制

十八、请简述设计一个 tcp 并发服务器的基本流程，并且将关键函数写出 （5 分）

socket(): 创建一个 socket 文件描述符
bind(): 绑定服务器的 ip 地址和端口号(以太网地址结构)
Listen(): 监听客户端
select(): 轮训所有监听的文件描述符
accept(): 等待客户端的连接

十九、请比较 select、poll 和 epoll （5 分）

select, poll, epoll 都是 IO 多路复用的机制，select poll 和 epoll 都是阻塞 IO

Select 和 poll 的缺点：

- （1）每次调用 select，都需要把 fd 集合从用户态拷贝到内核态，开销在 fd 很多时会很大
- （2）同时每次调用 select 都需要在内核遍历传递进来的所有 fd，开销在 fd 很多时也很大

二十、一个标准 C 语言项目的源代码目录结构为：src、include、lib，假如在 src 目录中有两个源文件分别为 main.c 和 list.c，头文件在 include 目录下，在 lib 目录下有一个动态库文件 libxyz.so.0.1，请问该如何处理该动态库文件才能使项目成功编译，请写出操作的过程。并且如果在 src 目录下进行编译，请写出编译的指令（编译器为 arm-linux-gcc） （5 分）

- 1) 创建软链接 `ln -s libxyz.so.0.1 libxyz.so`
- 2) `arm-linux-gcc main.c list.c -L./lib -lxyz`

二十一、请写出以下代码中各个变量分别存在于进程地址空间的哪一块区域

```
Int g_cnt;          g_cnt:  _____.bss_____
Int cnt = 10;       cnt:  _____.data_____
Static int sum;     sum:  _____.bss_____
Static int g_sum;   g_sum:  _____.bss_____
```

```
Void func(int x)    x:  ____ 栈_____
{
    Static int k = 10;    k:  _____.data_____
}
```

```
Void main()
{
    char *p;          p:  _____.栈_____
    p = (char *)malloc(10);
```

```

Strcpy(p, "hello"); "hello": ____将.rodata 上的 hello 拷贝至堆上____

Char ch[] = "hello"; ch: ____栈____ "hello": ____栈____

Char *str = "hello"; str: ____栈____ "hello": _____.rodata_____

}

```

二十二、用变量 a 给出下面的定义

a) 一个整型数

`Int a;`

b) 一个指向整型数的指针

`Int *a;`

c) 一个指向指针的指针，它指向的指针是指向一个整型数

`Int **a;`

d) 一个有 10 个整型数的数组

`Int a[10];`

e) 一个有 10 个指针的数组，该指针是指向一个整型数的

`Int *a[10];`

f) 一个指向有 10 个整型数数组的指针

`Int (*a)[10];`

g) 一个指向函数的指针，该函数有一个整型参数并返回一个整型数

`Int (*a)(int);`

h) 一个有 10 个指针的数组，该指针指向一个函数，该函数有一个整型参数并返回一个整型数

`Int (*a[10])(int);`