

Concurrent Programming

13 元培数科 常欣海

Here we go! →

2024/12/28



Emphasis

Notes

- Note Link

Homework Review

HW11

学号	HW-分数	HW-问题
2300012932	10	
2300012935	10	
2300012950	10	
2300012951	10	
2300013083	10	
2300013115	10	
2300013158	10	
2300013201	10	
2300013222	10	
2300013230	10	
2300013272	10	
2300017788	10	
2300094610	10	

Exercises

E1

20. 下列关于 C 语言中进程模型和线程模型的说法中，错误的是：
- A. 每个线程都有它自己独立的线程上下文，包括线程 ID、程序计数器、条件码、通用目的寄存器值等
 - B. 每个线程都有自己独立的线程栈，任何线程都不能访问其他对等线程的栈空间
 - C. 不同进程之间的虚拟地址空间是独立的，但同一个进程的不同线程共享同一个虚拟地址空间
 - D. 一个线程的上下文比一个进程的上下文小得多，因此线程上下文切换要比进程上下文切换快得多

E1

20. 下列关于 C 语言中进程模型和线程模型的说法中，错误的是：
- A. 每个线程都有它自己独立的线程上下文，包括线程 ID、程序计数器、条件码、通用目的寄存器值等
 - B. 每个线程都有自己独立的线程栈，任何线程都不能访问其他对等线程的栈空间
 - C. 不同进程之间的虚拟地址空间是独立的，但同一个进程的不同线程共享同一个虚拟地址空间
 - D. 一个线程的上下文比一个进程的上下文小得多，因此线程上下文切换要比进程上下文切换快得多

解析：B ↪

考察 12.3 12.4 线程模型，属于简单题。B 选项，不同的线程栈是不对其他线程设防的。所以，如果一个线程以某种方式得到一个指向其他线程栈的指针，那么它就可以读写这个栈的任何部分。 ↪

E2

22. 同一个进程内的不同线程共享以下几项？

- a) 函数体内部的静态变量
 - b) 堆
 - c) 文件描述符表
 - d) 条件码
- A. 1 B. 2 C. 3 D. 4

E2

22. 同一个进程内的不同线程共享以下几项？

- a) 函数体内部的静态变量
 - b) 堆
 - c) 文件描述符表
 - d) 条件码
- A. 1 B. 2 C. 3 D. 4

22. C。a) 是共享的（在数据段），b) 也是共享的（栈是私有的），c) 是共享的，属于书上原话，或者从同一进程的概念中读解出，d) 是特别的上下文，是私有的。所以有 3 个。

E3

24. 以下程序在输出有限行之后就终止了，请问最有可能的原因是（假定所有函数都正常执行）

```
#include "csapp.h"
void *thread(void *dummy){
    while (1){
        printf("hello, world!\n");
        Sleep(1);
    }
}
int main(){
    pthread_t tid;
    Pthread_create(&tid, NULL, thread, NULL);
    Sleep(3);
}
```

- A. 主线程结束必然引发所有对等线程结束
- B. 主线程结束时调用了_exit 导致进程结束
- C. 主线程结束后，内核发送 SIGKILL 杀死进程
- D. 主线程结束后，内核观察到对等线程运行时间过长，将其杀死

E3

24. 以下程序在输出有限行之后就终止了，请问最有可能的原因是（假定所有函数都正常执行）

```
#include "csapp.h"
void *thread(void *dummy){
    while (1){
        printf("hello, world!\n");
        Sleep(1);
    }
}
int main(){
    pthread_t tid;
    Pthread_create(&tid, NULL, thread, NULL);
    Sleep(3);
}
```

- A. 主线程结束必然引发所有对等线程结束
 - B. 主线程结束时调用了_exit 导致进程结束
 - C. 主线程结束后，内核发送 SIGKILL 杀死进程
 - D. 主线程结束后，内核观察到对等线程运行时间过长，将其杀死
24. B。注意主线程如果调用诸如 `pthread_exit` 的函数退出，是不会引发其他线程退出的，所以 A 为一错误描述。但是如果主线程调用了 `exit`（或者 `return`），则整个进程都会结束，自然其他线程也结束。

E4

16. 下列关于 Web 服务器向客户端提供动态内容（遵从 CGI 标准）的叙述中，哪一个是错误的？
- A. 客户端提出的 GET 请求的参数在 URI 中传递，并用“&”字符分隔文件名和多个参数
 - B. 接到客户端的请求，Web 服务器创建一个子进程执行相应的 CGI 程序
 - C. 通过设置环境变量，Web 服务器将 URI 中传递的参数传递给子进程
 - D. CGI 程序将动态内容重定向到与客户端关联的已连接描述符 connfd

E4

16. 下列关于 Web 服务器向客户端提供动态内容（遵从 CGI 标准）的叙述中，哪一个是错误的？
- A. 客户端提出的 GET 请求的参数在 URI 中传递，并用“&”字符分隔文件名和多个参数
 - B. 接到客户端的请求，Web 服务器创建一个子进程执行相应的 CGI 程序
 - C. 通过设置环境变量，Web 服务器将 URI 中传递的参数传递给子进程
 - D. CGI 程序将动态内容重定向到与客户端关联的已连接描述符 connfd

16. A。文件名和参数应该用 ? 分隔。

E5

19. 在以下基于线程的并发 Echo 服务器代码中，哪一个代码片段会潜在引发不正确的程序执行结果？

```
A. int main(int argc, char **argv)
{
    /*假设有序对的” */
    int listenfd, connfd;
    socklen_t clientlen;
    struct sockaddr_storage clientaddr;
    pthread_t tid;
B. listenfd = Open_listenfd(argv[1]);
C. while (1) {
    connfd = Accept(listenfd, (SA *) &clientaddr,
    &clientlen);
    Pthread_create(&tid, NULL, thread, &connfd);
}
D. void *thread(void *vargp)
{
    int connfd = *((int *)vargp);
    Pthread_detach(pthread_self());
    echo(connfd);
    Close(connfd);
    return NULL;
}
```

E5

19. 在以下基于线程的并发 Echo 服务器代码中，哪一个代码片段会潜在引发不正确的程序执行结果？

```
A. int main(int argc, char **argv)
{
    /*假设有序对的” */
    int listenfd, connfd;
    socklen_t clientlen;
    struct sockaddr_storage clientaddr;
    pthread_t tid;
    listenfd = Open_listenfd(argv[1]);
C. while (1) {
    connfd = Accept(listenfd, (SA *) &clientaddr,
    &clientlen);
    Pthread_create(&tid, NULL, thread, &connfd);
}
D. void *thread(void *vargp)
{
    int connfd = *((int *)vargp);
    Pthread_detach(pthread_self());
    echo(connfd);
    Close(connfd);
    return NULL;
}
```

19. C。否则会在对等线程的赋值和主线程的 accept 之间引入竞争，如果赋值在 accept 之后完成，描述符值就错了。

E6

12. 使用浏览器打开网页 www. pku. edu. cn 的过程中，下列网络协议中，可能会被用到的网络协议有____个

- ① DNS
- ② TCP
- ③ IP
- ④ HTTP

- A. 1
- B. 2
- C. 3
- D. 4

E6

12. 使用浏览器打开网页 www.pku.edu.cn 的过程中，下列网络协议中，可能会被用到的网络协议有____个

- ① DNS
- ② TCP
- ③ IP
- ④ HTTP

A. 1 B. 2 C. 3 D. 4

答案：D

考察对于网络的整体理解。按照书上涉及的内容，要解析域名必须用到 DNS 协议，要传输网页必须用到 HTTP 协议，HTTP 协议构建于 TCP/IP 协议之上，因此，所有的协议都可能会被用到。因为存在诸如 HTTPS，QUIC 等其他传输网页的协议，故加了可能两字。

E7

14. 下列关于进程与线程的描述中，哪一个是不正确的？
- A. 一个进程可以包含多个线程
 - B. 进程中的各个线程共享进程的代码、数据、堆和栈
 - C. 进程中的各个线程拥有自己的线程上下文
 - D. 线程的上下文切换比进程的上下文切换快

E7

14. 下列关于进程与线程的描述中，哪一个是不正确的？
- A. 一个进程可以包含多个线程
 - B. 进程中的各个线程共享进程的代码、数据、堆和栈
 - C. 进程中的各个线程拥有自己的线程上下文
 - D. 线程的上下文切换比进程的上下文切换快

答案：B

E8

15. 给定下列代码片段：

```
char **ptr; /* global var */
int main(int main, char *argv[])
{
    long i; pthread_t tid;
    char *msgs[2] = {"Hello from foo", "Hello from bar"};
    ptr = msgs;
    for (i = 0; i < 2; i++)
        Pthread_create(&tid, NULL, thread, (void *)i);
    Pthread_exit(NULL);

void *thread(void *vargp)
{
    long myid = (long)vargp;
    static int cnt = 0;
    printf("[%ld]: %s (cnt=%d)\n", myid, ptr[myid], ++cnt);
    return NULL;
}
```

下列哪一组变量集合是对等线程 1 引用的？

- A. ptr, cnt, i.m, msgs.m, myid.p0, myid.p1
- B. ptr, cnt, msgs.m, myid.p0
- C. ptr, cnt, msgs.m, myid.p1
- D. ptr, cnt, i.m, msgs.m, myid.p1

E8

15. 给定下列代码片段：

```
char **ptr; /* global var */
int main(int main, char *argv[])
{
    long i; pthread_t tid;
    char *msgs[2] = {"Hello from foo", "Hello from bar"};
    ptr = msgs;
    for (i = 0; i < 2; i++)
        Pthread_create(&tid, NULL, thread, (void *)i);
    Pthread_exit(NULL);

void *thread(void *vargp)
{
    long myid = (long)vargp;
    static int cnt = 0;
    printf("[%ld]: %s (cnt=%d)\n", myid, ptr[myid], ++cnt);
    return NULL;
}
```

下列哪一组变量集合是对等线程 1 引用的？

- A. ptr, cnt, i.m, msgs.m, myid.p0, myid.p1
- B. ptr, cnt, msgs.m, myid.p0
- C. ptr, cnt, msgs.m, myid.p1
- D. ptr, cnt, i.m, msgs.m, myid.p1

答案：C

Notices

THANKS

Made by WalkerCH

changxinhai@stu.pku.edu.cn

Reference: [Weicheng Lin]'s presentation.

Reference: [Arthals]'s templates and content.

