1、内存是进程范围or线程范围；\_\_\_\_

CPU调度时，针对进程or线程；\_\_\_\_

函数调用堆栈，针对进程or线程。\_\_\_\_

1. 写出内存拷贝函数memcpy的功能实现代码
2. 内存有哪些分配方式，他们的区别是什么？
3. char a[]=”hello”,

int b[100]={0}；

void\*p=malloc(sizeof(int)\*100)；

const char \*strstr(char str[100])

{

}

则下面结果是：

sizeof(a)=\_\_

sizeof(b)=\_\_\_

sizeof(p)=\_\_\_

sizeof(str)=\_\_\_.

1. 栈与堆的区别
2. 当你申请空间失败的时候，该怎么做
3. 程序里如何保证内存不泄露
4. ptr1 = (char \*)malloc(4);

ptr2= (char \*)malloc(100);

ptr3 = (char \*)malloc(400);

a = sizeof(ptr1);

b = sizeof(prt2);

c = sizeof(ptr3);问a = ，b = ，c =

9、以下程序的正确与否？并说明原因

1. void Memory\_0（char \*p）

{

p = (char \*)malloc(100);

}

1. Char \*Memory\_1(char \*p)

{

char p[] = “abcdefgh”;

return p;

}

1. void Memory\_3(char \*\*p)

{

\*p = (char \*\*)malloc(100);

}

1. 写一个安全的内存拷贝的函数
2. 以下哪句说法是正确的。

A 在页式存储管理中，用户应将自己的程序划分成若干个相等的页。

B 所有进程都挂起时，系统陷入死锁。

C 执行系统调用可以被中断。

D 进程优先数是进程调度的重要依据，必须根据进程运行情况动态改变。

1. 内存对齐指的是什么
2. 程序分析：

char g\_c[6];

int g\_i1;

int g\_i2 = 10;

int main()

{

Int l\_i = 89;

return 0;

}

请尝试说明。

1. 变量g\_c， g\_il， g\_i2, 以及 l\_i各自在a.out中的那些段，（text， data， bss…等等…）？
2. g\_c, g\_il, g\_i2, l\_i 各自运行在哪些地址空间中？

14、解答题

写出下面四个函数你认为有问题的地方，并说明原因

int tst\_malloc\_buf(char \*buf)

{

buf = (char \*)malloc(128);

if(buf != NULL)

return 0;

return -1;

}

int tst\_fill\_str(char \*buf)

{

if(buf != NULL)

{

sprintf(buf,”hello world!\n”);

return 0;

}

return -1;

}

int tst\_free\_buf(char \*buf)

{

if(buf != NULL)

{

free(buf);

buf = NULL;

}

return 0；

}

int main(int argc, char \*argv[])

{

char \*buf1 = malloc(128);

if(buf1 != NULL)

{

tst\_fill\_str(buf1);

printf(“%s\n”,buf1);

tst\_free\_buf(buf1);

}

char \*buf2 = “wuhanzhihuibweiishen”;

tst\_fill\_str(buf2);

printf(“%s”,buf2);

char \*buf3 = NULL;

if(tst­\_malloc\_buf(buf3) == 0)

{

tst\_fill\_str(buf3);

printf(“%s”,buf3);

}

if(buf 1 != NULL)

{

free（buf1）；

buf1 = NULL;

}

}

1. 自己编程实现memcpy函数，void \*memcpy(void \*dest, const void \*src, size\_t n)，注意不能产生数据覆盖;
2. 数据库存满了，会怎样？
3. memcpy 与 memmove 区别