**作业3**

3.1 fork、exec、wait等是进程操作的常用API，请调研了解这些API的使用方法。

（1）请写一个C程序，该程序首先创建一个1到10的整数数组，然后创建一个子进程，并让子进程对前述数组所有元素求和，并打印求和结果。等子进程完成求和后，父进程打印“parent process finishes”,再退出。

（2）在（1）所写的程序基础上，当子进程完成数组求和后，让其执行ls -l命令(注：该命令用于显示某个目录下文件和子目录的详细信息)，显示你运行程序所用操作系统的某个目录详情。例如，让子进程执行 ls -l /usr/bin目录，显示/usr/bin目录下的详情。父进程仍然需要等待子进程执行完后打印“parent process finishes”,再退出。

（3）请阅读XV6代码（<https://pdos.csail.mit.edu/6.828/2021/xv6.html>），找出XV6代码中对进程控制块（PCB）的定义代码，说明其所在的文件，以及当fork执行时，对PCB做了哪些操作？

提交内容

1. 所写C程序，打印结果截图，说明等
2. 所写C程序，打印结果截图，说明等
3. 代码分析介绍

3.2 请阅读以下程序代码，回答下列问题

（1）该程序一共会生成几个子进程？请你画出生成的进程之间的关系（即谁是父进程谁是子进程），并对进程关系进行适当说明。

（2）如果生成的子进程数量和宏定义LOOP不符，在不改变for循环的前提下，你能用少量代码修改，使该程序生成LOOP个子进程么？

提交内容

1. 问题解答，关系图和说明等
2. 修改后的代码，结果截图，对代码的说明等

#include<unistd.h>

#include<stdio.h>

#include<string.h>

#define LOOP 2

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

{

pid\_t pid;

int loop;

for(loop=0;loop<LOOP;loop++) {

if((pid=fork()) < 0)

fprintf(stderr, "fork failed\n");

else if(pid == 0) {

printf(" I am child process\n");

}

else {

sleep(5);

}

}

return 0;

}