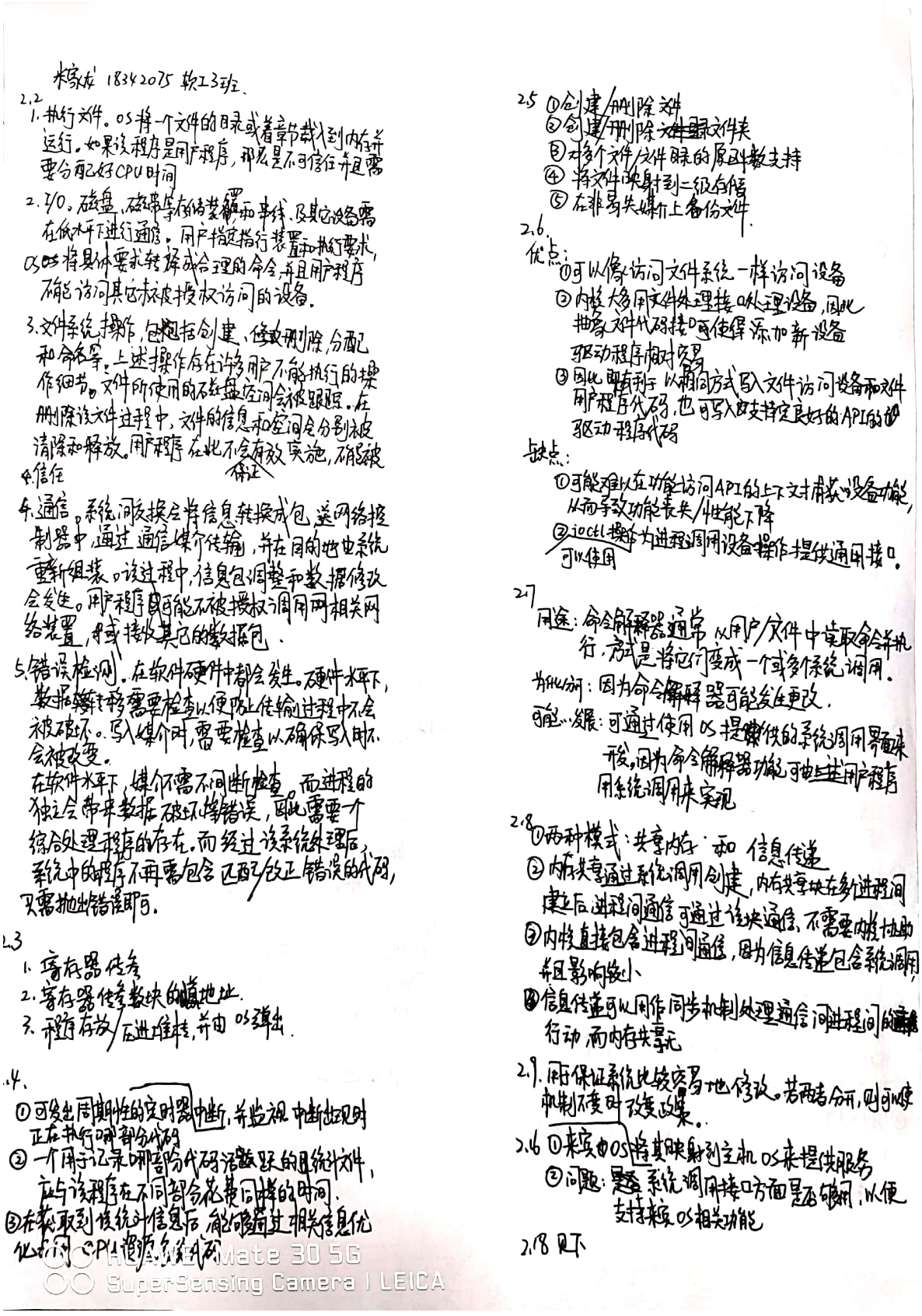
操作系统第一次书面作业

- 数据科学与计算机学院
- 软工三班
- 米家龙
- 18342075



```c
// 在 linux 上写
#include <stdio.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>

#define BUFFLENGTH 8192

int main(int argc, char **argv)
{
  if (argc != 3) // 参数不够
  {
    printf("usage: copy src dst\n");
    return -1;
  }
  int src, dst;
  char buffer[BUFFLENGTH];
  src = open(argv[1], O_RDONLY);
  dst = open(argv[2],
             O_RDWR | O_CREAT | O_TRUNC,   // 设置权限
             S_IRUSR | S_IWUSR | S_IXUSR); // 设置用户权限
  int tmp;
  while ((tmp = read(src, buffer, BUFFLENGTH)) > 0) // 不断读取
  {
    if (write(dst, buffer, tmp) != tmp) // 写入失败
      printf("ERROR in writing\n");
  }
  if (tmp < 0) // 读取失败
    printf("ERROR in reading\n");

  // 关闭文件
  close(src);
  close(dst);
  return 0;
}
```

林宇龙 18342075 软工3班.

3.1: 主要在于执行频率
①短期调度须经常调用新进程
②中期调度将部分程序移出内存,在中断位置继续
③长期调度则不经常不频繁调用,可能在进程离开系统时才会唤起.

3.2
①保存正在运行的进程状态,主要包括CPU寄存器的值和内存分配,~~上下文分配~~
②恢复进程的状态,上下文切换,进行数据刷新和指令缓存等操作。

3.3
①PRC不支持"最多一次"、"至少一次"的语义,则PRC服务器则不会保证不会多次调用远程过程,而远程过程的单次调用可能导致服务器多次退出
②为了支持上述语义,需要服务器维护客户端状态。如果不能支持,则该系统只能安全提供不会更改数据或提供时间敏感的远程过程。

.5
①对称和非对称通信.
对称通信允许收发者之间有1个集合点。但在阻塞发送时并不需要集合点,并且消息是只能同步传递的。因此需要两种使同步形式
②自动和显式缓冲.
自动缓冲提供了无限长的队列,保证发送者复制消息时不会阻塞,但缺点是会浪费内存。可通过明确缓冲区大小来避免浪费.
③复制发送和引用发送.
复制发送不允许接收者改变参数状态,而引用发送允许。因此引用发送允许程序员写一个分布式版本的集中的应用程序
④固定大小和可变大小消息
固定大小的消息在特定大小缓冲区中可容纳数量已知,而可变大小消息则未知.

4.3
内核线程页面错误时,另外的内核线程会转换为使用交错时间。对于单一线程进程,如果发生页错误,其性能会下降。

4.4 共享:堆内存和全局变量
独立:寄存器值和栈内存.

4.6.①如果系统将进程和线程视为相似实体,系统代码可简化
②上述情况会使得进程资源限制更加困难,因为需要线程与进程哪些一致和执行重复计数.

4.7 ①C: 5
②P: 0

补充

2.11
①虚存子系统和存储子系统相耦合,许多OS允许文件被映射到一个执行进程的虚存空间
②虚存子系统通常用存储子系统提供当前不在内存的页
③虚存中更新的文件在磁盘刷新前缓冲到物理内存。

2.12 优点: ①增加新服务不需要修改内核.
②在用户模式中比在内核模式中更安全和易操作
③OS可能更可靠
相互作用方式:使用进程间通信机制
缺点:进程间通信过度联系,为了保证用户程序和系统服务相互作用会频繁使用操作系统的消息传递.

2.13
①模块化内核方法要求子系统通过创建接口来相互作用
②两者细节相似,但分层内核要有严格排序的子系统,从而限制低层次不允许引用高层次子系统。而模块化内核则彼此之间可以随意调用