

计算机操作系统原理

实验报告

班 级：1504201

姓 名：殷悦

学 号：150120526

实验编号：实验二

序言

为了培养学生的实践动手能力，帮助学生体会与验证所学的知识，设计该实验体系。课内实践安排 8 学时，其余需求学生可以课下完成。

实验报告基本要求：

1. 学生提交各个实验程序的流程图，并填入实验报告中；
2. 实验报告附上运行结果的截图；
3. 对实验的结果进行分析，写出自己的分析结果；
4. 实验的源代码、可执行代码和实验报告需要打包交到班级统一刻盘提交保存；个人文件的命名以个人的姓名+学号命名；班级的以班级编号命名。
5. 程序可视性要好，要有试验程序证明作品的正确性。
6. 所有的实验报告采用 A4 打印，字体字号遵循模版规定。
7. 源程序要加注释，要有测试数据及结果。
8. 每个实验的报告要另起一页。

基本要求：每个学生必须完成前 4 个实验，实验结果将逐个检查，通过者，方可提交实验报告，并获得实验成绩。

有精力的同学可以额外选作其他实验，额外多做的实验根据完成质量可以酌情加分 1-5 分。

报告的封面格式以及内容的格式不要自己调整。

实验二 常规存储器管理

一、实验目的

1. 掌握内存管理的基本功能和分区法内存分配的基本原理。
2. 学会 windows 操作系统下使用 c 语言函数和系统调用进行编程的方法。
3. 利用 c 语言设计实现分区法内存分配算法。
4. 验证无虚存的存储管理机制。

二、实验要求

1. 学生应完成如下章节的学习：进程和线程、调度、存储管理。
2. 在操作系统平台上，使用 c 语言编程，调用相关系统调用进行设计实现。

三、实验内容

1. 创建空闲存储管理表和模拟内存。
2. 设计并实现一个内存分配程序，分配策略可以分别采用最先适应算法、最佳适应算法和最坏适应算法等，并评价不同分配算法的优劣。
3. 提供一个用户界面，利用它用户可输入不同的分配策略。
4. 进程向内存管理程序发出申请、释放指定数量的内存请求，内存管理程序调用对应函数，响应请求。

四、实验方案指导

该实验方案由以下几个关键设计项目组成。

1. 设计实现一个空闲分区表。
2. 设计实现模拟内存。

考虑实现的便利，本方案不访问真正的内存。定义一个字符数组 `char mm[mem_size]` 或使用 Linux 系统调用 `mm=malloc(mem_size)`，用来模拟内存。

利用指针对模拟内存进行访问。

3. 设计一组管理物理内存空间的函数。用户接口由以下三个函数组成：

```
void*mm_request(int n)
```

申请 `n` 个字节的内存空间。如申请成功，则返回所分配空间的首地址；如不能满足申请，则返回空值。

```
void mm_release(void*p)
```

释放先前申请的内存。如果释放的内存与空闲区相邻，则合并为一个大空闲区；如果与空闲区不相邻，则成为一个单独的空闲区。

```
void*mm_init(int mem_size)
```

内存初始化。返回 mm 指针指向的空闲区。

4. 设计实现不同策略的内存分配程序。

对于采用不同分配策略的内存管理程序，从以下两个方面进行调度程序性能的比对：内存利用率以及找到一个合适的分配空间所需查找的步骤。

设置一个模拟实验。分别构建一个随机生成的请求与释放队列。释放队列中的操作总是得到满足，队列总为空；请求队列的操作能否被满足，取决于空闲区能否满足申请的空间大小。若不能满足，则该操作在队列中等待相应释放操作唤醒。请求队列采用 FIFO 管理，以避免饥饿现象的发生。

内存管理程序应对内存初始化，随机设定内存空间的占有、空闲情况，随机设定申请和释放的操作队列。调用释放操作开始运行，调用申请操作，如能满足，则分配空间，否则等待释放操作唤醒。

下面给出一个模拟内存管理的程序框架（伪码形式）。可对性能数据指标进行统计。

```
=====
for(i=0;i<sim_step;i++){                               /*设定模拟程序执行次数*/
    do{                                                    /*循环调用请求操作，直到请求不成功
*/
        get size n of next request;                       /*设定请求空间大小*/
        mm_request(n);                                     /*调用请求操作*/
    }while(request successful);                             /*请求成功，循环继续*/
    record memory utilization;                             /*统计内存使用率*/
    select block p to be release;                          /*释放某空间 p*/
    release(p);                                             /*调用释放操作*/
}
=====
```

以上程序由主循环控制固定次数的模拟步骤。每次循环，程序完成如下处理步骤：内循环尽可能多地满足内存请求，请求内存大小值随机生成。一旦请求失败，挂起内存管理程序，直至释放操作被执行。此时进行系统内存利用率的统计、计算，随机挑选一个内存分配空间完成释放操作。本次主循环执行完毕，开始下

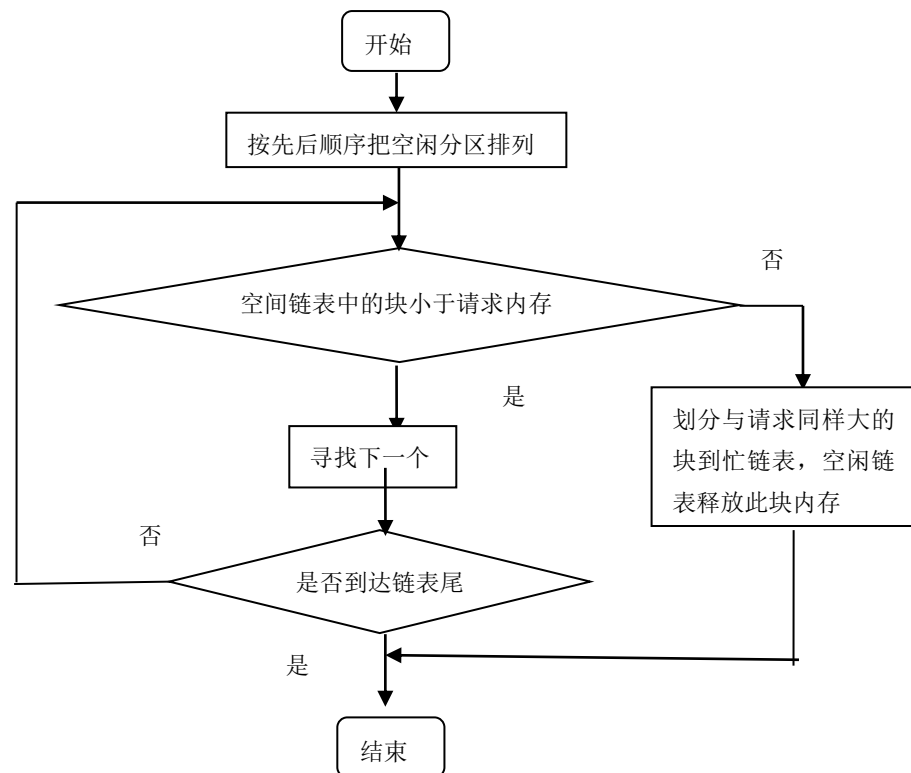
一次循环。

需要在程序中完成以下设计：确定请求分配空间大小，统计性能数据，选择一个内存区释放。

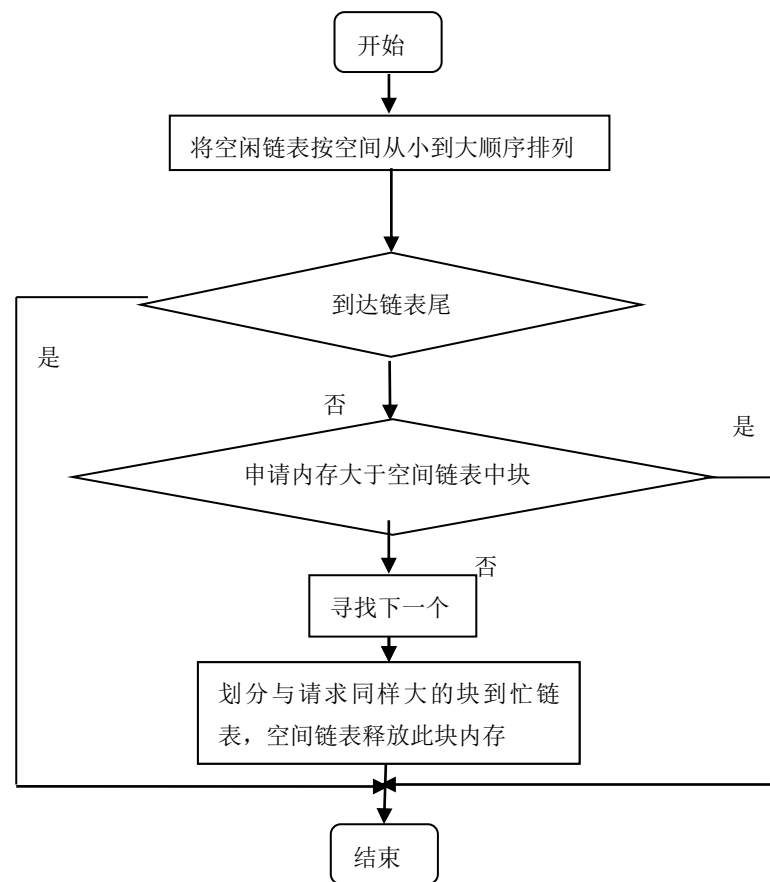
以下部分由学生填写：

1. 程序流程图

(1) 最先适应算法流程图



(2) 最佳适应算法流程图。



(3)

(4) 最坏适应算法流程图。

2. 实验结果

(1) 分区运行结果图

(2) 分区成功后运行结果图

```

2股悦_操作系统实验二 — a.out — 66x17
[YinYue:2股悦_操作系统实验二 yy$ gcc mem_opr.c
YinYue:2股悦_操作系统实验二 yy$ ./a.out
1.最先适应      2.最佳适应      3.最坏适应
>1
内存从 0x7fd8cc000000到 0x7fd8cc000bb8内存总大小:3000空闲分区:
Start      End      Size
0x7fd8cc000001 0x7fd8cbffffd34 714
0x7fd8cc000116 0x7fd8cbffffd9b 611
0x7fd8cc00002d 0x7fd8cbffffdf8 518
0x7fd8cc0001f9 0x7fd8cbffffeb2 332
已用分区:
PID      Start      End      Size
3        0x7fd8cc000207 0x7fd8cbffffe06 504
2        0x7fd8cc000264 0x7fd8cbfffffd2 44
1        0x7fd8cc0002cb 0x7fd8cbffffee9 277
r:申请内存      f:释放内存      q:退出

```

```
2股悦_操作系统实验二 — a.out — 66x17
[YinYue:2股悦_操作系统实验二 yy$ ./a.out
1.最先适应      2.最佳适应      3.最坏适应
>2
内存从0x7fd69c800000到0x7fd69c800bb8内存总大小:3000空闲分区:
Start      End      Size
0x7fd69c8002d4 0x7fd69c7ffff90 110
0x7fd69c800109 0x7fd69c7ffff18 230
0x7fd69c800135 0x7fd69c7fffee9 277
0x7fd69c800001 0x7fd69c7ffe0b 499
已用分区:
PID      Start      End      Size
4        0x7fd69c8000e7 0x7fd69c7ffd2b 723
3        0x7fd69c800116 0x7fd69c7ffef6 264
2        0x7fd69c8001d7 0x7fd69c7ffeca 308
1        0x7fd69c8001f4 0x7fd69c7fff87 119
r:申请内存      f:释放内存      q:退出
```

```
2股悦_操作系统实验二 — a.out — 62x18
[YinYue:2股悦_操作系统实验二 yy$ ./a.out
1.最先适应      2.最佳适应      3.最坏适应
>3
内存从0x7fec9a0000000到0x7fec9a000bb8内存总大小:3000空闲分区:
Start      End      Size
0x7fec9a000012d 0x7fec9ffffd7e 640
0x7fec9a0000118 0x7fec9ffffd86 632
0x7fec9a000003e 0x7fec9ffffee0 286
0x7fec9a0000001 0x7fec9fffff80 126
已用分区:
PID      Start      End      Size
5        0x7fec9a0000279 0x7fec9ffffef4 266
4        0x7fec9a0000281 0x7fec9ffffee7 279
3        0x7fec9a000011f 0x7fec9ffffed2 300
2        0x7fec9a0000112 0x7fec9fffffc1 61
1        0x7fec9a000007f 0x7fec9fffff75 137
r:申请内存      f:释放内存      q:退出
```

(3) 为分区申请资源运行结果图

```
2股悦_操作系统实验二 — a.out — 61x34
r:申请内存      f:释放内存      q:退出
进程5申请内存成功空闲分区:
Start           End           Size
0x7fbb2c800001  0x7fbb2c7ffffb  3
0x7fbb2c8002a2  0x7fbb2c7ffe0f  223
0x7fbb2c8000ec  0x7fbb2c7ffde8  534
0x7fbb2c80006e  0x7fbb2c7ffd25  729
已用分区:
PID      Start           End           Size
5        0x7fbb2c80023d  0x7fbb2c8002a1  100
4        0x7fbb2c8002da  0x7fbb2c7ffe0f  495
3        0x7fbb2c800217  0x7fbb2c7fff91  109
2        0x7fbb2c800144  0x7fbb2c7fff13  235
1        0x7fbb2c800004  0x7fbb2c7ffdc2  572
r:申请内存      f:释放内存      q:退出
r 300
错误输入
r:申请内存      f:释放内存      q:退出
进程6申请内存成功空闲分区:
Start           End           Size
0x7fbb2c800001  0x7fbb2c7ffffb  3
0x7fbb2c8002a2  0x7fbb2c7ffe0f  223
0x7fbb2c800219  0x7fbb2c7ffde8  234
0x7fbb2c80006e  0x7fbb2c7ffd25  729
已用分区:
PID      Start           End           Size
6        0x7fbb2c8000ec  0x7fbb2c800218  300
5        0x7fbb2c80023d  0x7fbb2c8002a1  100
4        0x7fbb2c8002da  0x7fbb2c7ffe0f  495
3        0x7fbb2c800217  0x7fbb2c7fff91  109
2        0x7fbb2c800144  0x7fbb2c7fff13  235
1        0x7fbb2c800004  0x7fbb2c7ffdc2  572
r:申请内存      f:释放内存      q:退出
```

(4) 最先适应算法，为分区申请 1 个单位的资源运行结果图


```
2股悦_操作系统实验二 — a.out — 61x32
[YinYue:2股悦_操作系统实验二 yy$ ./a.out
1.最先适应      2.最佳适应      3.最坏适应
>2
内存从 0x7f8fcb000000到 0x7f8fcb000bb8内存总大小:3000空闲分区:
Start          End          Size
0x7f8fcb000172 0x7f8fcaffffec1 317
0x7f8fcb00014e 0x7f8fcaffffe0e 496
0x7f8fcb000001 0x7f8fcaffffdda 548
已用分区:
PID    Start          End          Size
4      0x7f8fcb0001f1 0x7f8fcaffff53 171
3      0x7f8fcb00013e 0x7f8fcaffffeb1 333
2      0x7f8fcb000196 0x7f8fcaffffe8d 369
1      0x7f8fcb000225 0x7f8fcaffffe95 361
r:申请内存      f:释放内存      q:退出
r 100
错误输入
r:申请内存      f:释放内存      q:退出
进程5申请内存成功空闲分区:
Start          End          Size
0x7f8fcb0001d7 0x7f8fcaffffec1 217
0x7f8fcb00014e 0x7f8fcaffffe0e 496
0x7f8fcb000001 0x7f8fcaffffdda 548
已用分区:
PID    Start          End          Size
5      0x7f8fcb000172 0x7f8fcb0001d6 100
4      0x7f8fcb0001f1 0x7f8fcaffff53 171
3      0x7f8fcb00013e 0x7f8fcaffffeb1 333
2      0x7f8fcb000196 0x7f8fcaffffe8d 369
1      0x7f8fcb000225 0x7f8fcaffffe95 361
r:申请内存      f:释放内存      q:退出
```

(5) 最先适应算法，为分区申请 1 个单位的资源运行结果图

```
YinYue:2股悦_操作系统实验二 yy$ ./a.out
1.最先适应      2.最佳适应      3.最坏适应
>1
内存从0x7fbb2c800000到0x7fbb2c800bb8内存总大小:3000空闲分区:
Start      End      Size
0x7fbb2c800001 0x7fbb2c7ffffb 3
0x7fbb2c80023d 0x7fbb2c7ffe0f 323
0x7fbb2c8000ec 0x7fbb2c7ffde8 534
0x7fbb2c80006e 0x7fbb2c7ffd25 729
已用分区:
PID      Start      End      Size
4      0x7fbb2c8002da 0x7fbb2c7ffe0f 495
3      0x7fbb2c800217 0x7fbb2c7fff91 109
2      0x7fbb2c800144 0x7fbb2c7fff13 235
1      0x7fbb2c800004 0x7fbb2c7ffdc2 572
r:申请内存      f:释放内存      q:退出
r 100
错误输入
r:申请内存      f:释放内存      q:退出
进程5申请内存成功空闲分区:
Start      End      Size
0x7fbb2c800001 0x7fbb2c7ffffb 3
0x7fbb2c8002a2 0x7fbb2c7ffe0f 223
0x7fbb2c8000ec 0x7fbb2c7ffde8 534
0x7fbb2c80006e 0x7fbb2c7ffd25 729
已用分区:
PID      Start      End      Size
5      0x7fbb2c80023d 0x7fbb2c8002a1 100
4      0x7fbb2c8002da 0x7fbb2c7ffe0f 495
3      0x7fbb2c800217 0x7fbb2c7fff91 109
2      0x7fbb2c800144 0x7fbb2c7fff13 235
1      0x7fbb2c800004 0x7fbb2c7ffdc2 572
r:申请内存      f:释放内存      q:退出
```

(6) 最先适应算法，继续为分区申请资源失败运行结果图

```
2股悦_操作系统实验二 — -bash — 61x32
0x7fbb2c8002a2 0x7fbb2c7ffe0f 223
0x7fbb2c8000ec 0x7fbb2c7ffde8 534
0x7fbb2c80006e 0x7fbb2c7ffd25 729
已用分区：
PID      Start          End            Size
5        0x7fbb2c80023d 0x7fbb2c8002a1 100
4        0x7fbb2c8002da 0x7fbb2c7ffe0f 495
3        0x7fbb2c800217 0x7fbb2c7fff91 109
2        0x7fbb2c800144 0x7fbb2c7fff13 235
1        0x7fbb2c800004 0x7fbb2c7ffdc2 572
r:申请内存      f:释放内存      q:退出
r 300
错误输入
r:申请内存      f:释放内存      q:退出
进程6申请内存成功空闲分区：
Start          End            Size
0x7fbb2c800001 0x7fbb2c7ffffb 3
0x7fbb2c8002a2 0x7fbb2c7ffe0f 223
0x7fbb2c800219 0x7fbb2c7ffde8 234
0x7fbb2c80006e 0x7fbb2c7ffd25 729
已用分区：
PID      Start          End            Size
6        0x7fbb2c8000ec 0x7fbb2c800218 300
5        0x7fbb2c80023d 0x7fbb2c8002a1 100
4        0x7fbb2c8002da 0x7fbb2c7ffe0f 495
3        0x7fbb2c800217 0x7fbb2c7fff91 109
2        0x7fbb2c800144 0x7fbb2c7fff13 235
1        0x7fbb2c800004 0x7fbb2c7ffdc2 572
r:申请内存      f:释放内存      q:退出
r 800
没有剩余空间，加入请求队列中
r:申请内存      f:释放内存      q:退出
```

(7) 最佳适应算法，为分区申请 1 个单位的资源运行结果图

```
2殷悦_操作系统实验二 — a.out — 62x32
[YinYue:2殷悦_操作系统实验二 yy$ ./a.out
1.最先适应      2.最佳适应      3.最坏适应
>2
内存从0x7fbe27003000到0x7fbe27003bb8内存总大小:3000空闲分区:
Start      End      Size
0x7fbe2700328d 0x7fbe27002f03 251
0x7fbe27003048 0x7fbe27002eff 255
0x7fbe27003001 0x7fbe27002e49 437
已用分区:
PID      Start      End      Size
4      0x7fbe27003100 0x7fbe27002e8c 370
3      0x7fbe270030fc 0x7fbe27002fb7 71
2      0x7fbe27003124 0x7fbe27002d72 652
1      0x7fbe270031b6 0x7fbe27002d5d 673
r:申请内存      f:释放内存      q:退出
r 100
错误输入
r:申请内存      f:释放内存      q:退出
进程5申请内存成功空闲分区:
Start      End      Size
0x7fbe270032f2 0x7fbe27002f03 151
0x7fbe27003048 0x7fbe27002eff 255
0x7fbe27003001 0x7fbe27002e49 437
已用分区:
PID      Start      End      Size
5      0x7fbe2700328d 0x7fbe270032f1 100
4      0x7fbe27003100 0x7fbe27002e8c 370
3      0x7fbe270030fc 0x7fbe27002fb7 71
2      0x7fbe27003124 0x7fbe27002d72 652
1      0x7fbe270031b6 0x7fbe27002d5d 673
r:申请内存      f:释放内存      q:退出
```

(8) 最坏适应算法，为分区申请 1 个单位的资源运行结果图结果分析

```
2股悦_操作系统实验二 — a.out — 67x36
[YinYue:2股悦_操作系统实验二 yy$ ./a.out
1.最先适应      2.最佳适应      3.最坏适应
>3
内存从0x7feca0000000到0x7feca0000bb8内存总大小:3000空闲分区:
Start      End      Size
0x7feca000012d 0x7fec9ffffd7e 640
0x7feca0000118 0x7fec9ffffd86 632
0x7feca000003e 0x7fec9ffffee0 286
0x7feca0000001 0x7fec9fffff80 126
已用分区:
PID      Start      End      Size
5        0x7feca0000279 0x7fec9ffffef4 266
4        0x7feca0000281 0x7fec9ffffee7 279
3        0x7feca000011f 0x7fec9ffffed2 300
2        0x7feca0000112 0x7fec9fffffc1 61
1        0x7feca000007f 0x7fec9fffff75 137
r:申请内存      f:释放内存      q:退出
r 5
错误输入
r:申请内存      f:释放内存      q:退出
进程6申请内存成功空闲分区:
Start      End      Size
0x7feca0000133 0x7fec9ffffd7e 635
0x7feca0000118 0x7fec9ffffd86 632
0x7feca000003e 0x7fec9ffffee0 286
0x7feca0000001 0x7fec9fffff80 126
已用分区:
PID      Start      End      Size
6        0x7feca000012d 0x7feca0000132 5
5        0x7feca0000279 0x7fec9ffffef4 266
4        0x7feca0000281 0x7fec9ffffee7 279
3        0x7feca000011f 0x7fec9ffffed2 300
2        0x7feca0000112 0x7fec9fffffc1 61
1        0x7feca000007f 0x7fec9fffff75 137
r:申请内存      f:释放内存      q:退出
```

3. 结果分析

- (1) 最先适应算法: 遍历整个空闲分区空间, 如果找到第一个满足的空闲分区, 就给申请者, 可能会浪费一定空间
- (2) 最佳适应算法: 按空间剩余大小, 先小后大的顺序查找空闲分区, 可找到最佳的空闲区域, 分配很合理, 花费时间较第一个更长
- (3) 最坏适应算法: 按照空间剩余大小, 先大后小的顺序查找空闲分区, 分配后剩余的空间仍旧很大, 减少了产生碎片的几率, 花费时间较第一个长, 较第二个短