

四川大学计算机学院、软件学院

实验报告

组长：史臻泽 组员：唐克旭、王鑫宇、封欢欢、张润发 专业：计算机科学与技术 班级：行政三班 12-13 周

课程名称	操作系统课程设计	实验课时	4 小时
实验项目	内存管理实验	实验时间	第 12 周到第 13 周
实验目的	1) 了解 Windows 及 Linux 内存管理机制。 2) 掌握页面虚拟存储技术。 3) 了解内存分配原理，特别是以页面为单位的虚拟内存分配方法。 4) 学会使用 Windows 下内存管理的基本 API 函数。 5) 了解进程中内存分配与虚内存的区别；		
实验环境	Clion2024.1.1		
实验内容（算法、程序、步骤和方法）	<p>实验内容：</p> <p>实验一：用一段程序获得当前系统的存储空间使用概况</p> <p>代码：</p> <pre>1 #include <iostream> 2 #include <windows.h> 3 4 using namespace std; 5 6 int main(int argc, char const *argv[]) 7 { 8 // system("chcp 65001"); 9 MEMORYSTATUSEX help1; 10 help1.dwLength = sizeof(help1); 11 GlobalMemoryStatusEx(&help1); 12 cout<<"当前整体存储如下："<<endl; 13 14 cout <<"物理内存总量："<<help1.ullTotalPhys/(1024 * 1024)<<"MB"<<endl; 15 cout <<"可用物理内存："<<help1.ullAvailPhys/(1024 * 1024)<<"MB"<<endl; 16 cout <<"页文件总量："<<help1.ullTotalPageFile/(1024 * 1024)<<"MB"<<endl; 17 cout <<"可用页文件："<<help1.ullAvailPageFile/(1024 * 1024)<<"MB"<<endl; 18 cout <<"虚拟内存总量："<<help1.ullTotalVirtual/(1024 * 1024)<<"MB"<<endl; 19 cout <<"可用虚拟内存："<<help1.ullAvailVirtual/(1024 * 1024)<<"MB"<<endl; 20 cout <<"额外虚拟内存："<<help1.ullAvailExtendedVirtual/(1024 * 1024)<<"MB"<<endl; 21 cout <<"物理内存负载情况："<<help1.dwMemoryLoad<<"%"<<endl; 22 return 0; 23 }</pre> <p>实验二（选做）：编写分页存储管理的程序。</p> <p>要求：</p> <p>(1) 给出初态；</p>		

(2) 程序入口参数:

分配时, 参数为进程名及请求分配的内存空间大小----字节数或页数。

回收时, 参数为进程名。

(3) 每次分配或回收后, 显示页表等的内容, 无法分配时, 给出回应信息。

函数:

main 函数:

```
int main(int argc, char const *argv[])
{
    init();
    while (1){
        if (!table())
        {
            break;
        }
    }
    return 0;
}
```

table () 选项列表:

```
bool table() {
    cout<<"1.创建进程"<<endl;
    cout<<"2.结束进程"<<endl;
    cout<<"3.查看内存"<<endl;
    cout<<"4.退出程序"<<endl;
    cout<<"请输入你要进行的操作";
    int num;
    cin>>num;
    switch (num) {
        case 1:{
            CreateProcess();
            break;
        }
        case 2:{
            DeleteProcess();
            break;
        }
        case 3:{
```

```

        show();
        break;
    }
    case 4:{
        return false;
    }
    default:{
        cout<<"你输入了错误选项"<<endl;
    }
}
return true;
}

```

创建进程函数：

```

bool CreateProcess(){
    cout<<"请输入进程号(0-99)，进程别名和所需内存大小，用空格隔开：";
    int num ,need;
    string name ;
    cin>>num>>name>>need;
    int help = CreateProcessCheck(num ,name , need);
    while (help == 0)
    {
        cout<<"请输入进程号(0-99)，进程别名，用空格隔开，输入-1 取消： ";
        cin>>num;
        if (num == -1)
        {
            help = -1;
            cout<<"已取消"<<endl;
            return false;
        }
        cin>>name>>need;
        help = CreateProcessCheck(num , name , need);
    }
    if (help == -1 )    return false;
    UsedPage += need;
    processtotal[num].name = name ;
    processtotal[num].NeedPage = need;
    cout<<"创建成功"<<endl;
    TotalProcess++;
    show();
    return true;
}

```

辅助判断创建条件是否合法：

```

int CreateProcessCheck(int num,string name ,int need){
    if (processtotal[num].NeedPage != -1)
    {
        cout<<"进程号被占用，请重试"<<endl;
        return 0;
    }
    if (need > (TotalPage - UsedPage)){
        if (need > 100)
        {
            cout<<"剩余内存不足，无法进行"<<endl;
            return -1;
        }

        cout<<"剩余内存不足，无法进行"<<endl;
        return 0;
    }
    for (int i = 0; i < 100; i++) {
        if (processtotal[i].name == name)
        {
            cout<<"名称重复，请重试"<<endl;
            return 0;
        }
    }
    return 1;
}

```

进程删除函数：

```

void DeleteProcess(){
    cout<<"1.通过程序号删除"<<endl;
    cout<<"2.通过名称删除"<<endl;
    int num;
    cin>>num;
    switch (num)
    {
        case 1:{
            cout<<"请输入程序号"<<endl;
            int num1;
            cin>>num1;
            if (processtotal[num1].NeedPage != -1)
            {
                cout<<"已删除"<<num1<<"号程序"<<endl;
                TotalProcess--;
                UsedPage -= processtotal[num1].NeedPage;
                processtotal[num1].NeedPage = -1;
                processtotal[num1].name = "";
                show();
            }
        }
    }
}

```

```

else{
    cout<<"无此程序号程序"<<endl;
}
break;
}
case 2:{
    cout<<"请输入程序名"<<endl;
    string name;
    cin>>name;
    bool flag = false;
    for (int i = 0; i < 100; i++)
    {
        if (processtotal[i].name == name)
        {
            cout<<"已删除"<<name<<"程序"<<endl;
            TotalProcess--;
            processtotal[i].name = "";
            UsedPage -= processtotal[i].NeedPage;
            processtotal[i].NeedPage = -1;
            flag = true;
            show();
            break;
        }
    }
    if (!flag)
    {
        cout<<"未找到该名称的程序"<<endl;
    }
    break;
}
default:{
    cout<<"你输入了错误的选项"<<endl;
    break;
}
}
}

```

状态查看函数:

```

void show() {
    cout<<"总内存量: "<<TotalPage<<"，已使用内存量: "<<UsedPage<<"，剩余内存量: "<<TotalPage - UsedPage<<"，进程数: "<<TotalProcess<<endl;
    cout<<"已使用的内存块为: ";
    cout<<pages[0];
    for (int i = 1; i < UsedPage; i++)
    {
        cout<<" "<<pages[i];
    }
}

```

```

    }
    cout<<endl;
}

```

初始化分配函数:

```

void init(){
    TotalPage = 100;
    UsedPage = 10;
    TotalProcess = 0;
    for (int i = 0; i < 100; i++)
    {
        processtotal[i].name = "";
        processtotal[i].NeedPage = -1;
        pages[i] = i;
    }
    random_device rd;
    mt19937 g(rd());
    shuffle(pages.begin(), pages.end(), g);
    show();
}

```

(接上)
实验内容(算法、程序、
步骤和方法)

其他头文件以及数据结构:

```

1  #include <iostream>
2  #include <algorithm>
3  #include <random>
4  #include <vector>
5
6  using namespace std;
7
8  int TotalPage;
9  int UsedPage;
10 int TotalProcess;
11
12 struct process
13 {
14     string name;
15     int NeedPage;
16 };
17
18 process processtotal[100];
19 vector<int> pages(n:100);

```

<div data-bbox="272 1088 384 1164" data-label="Text"><p>数据记录 和计算</p></div>	<div data-bbox="504 215 754 255" data-label="Section-Header"><p>实验一运行结果：</p></div> <div data-bbox="504 280 1331 813" data-label="Code-Block"><pre>当前整体存储如下： 物理内存总量：16167MB 可用物理内存：4683MB 页文件总量：19367MB 可用页文件：6291MB 虚拟内存总量：134217727MB 可用虚拟内存：134213551MB 额外虚拟内存：0MB 物理内存负载情况：71% Process finished with exit code 0</pre></div> <div data-bbox="504 817 746 857" data-label="Section-Header"><p>实验二运行结果：</p></div> <div data-bbox="504 875 1428 1462" data-label="Code-Block"><pre>总内存量：100，已使用内存量：10，剩余内存量：90，进程数：0 已使用的内存块为：56,47,1,16,0,73,45,4,9,88 1.创建进程 2.结束进程 3.查看内存 4.退出程序 请输入你要进行的操作1 请输入进程号(0-99)，进程别名和所需内存大小，用空格隔开：0 进程1 20 创建成功 总内存量：100，已使用内存量：30，剩余内存量：70，进程数：1 已使用的内存块为：56,47,1,16,0,73,45,4,9,88,83,2,80,48,30,53,46,68,50,19,34,85,42,98,27,38,18,17,23,11 1.创建进程 2.结束进程 3.查看内存 4.退出程序 请输入你要进行的操作1 请输入进程号(0-99)，进程别名和所需内存大小，用空格隔开：0 进程2 10 进程号被占用，请重试 请输入进程号(0-99)，进程别名，用空格隔开，输入-1取消：1 进程1 10 名称重复，请重试 请输入进程号(0-99)，进程别名，用空格隔开，输入-1取消：1 进程2 80 剩余内存不足，无法进行 请输入进程号(0-99)，进程别名，用空格隔开，输入-1取消：80 进程2 50 创建成功</pre></div> <div data-bbox="504 1485 1428 1762" data-label="Code-Block"><pre>创建成功 总内存量：100，已使用内存量：80，剩余内存量：20，进程数：2 已使用的内存块为：56,47,1,16,0,73,45,4,9,88,83,2,80,48,30,53,46,68,50,19,34,85,42,98,27,38,18,17,23,11,33,35,8,72,77,67,71,54,96,64,62,69,84,58,26,59,75,13,5,70,81,97,49,15,78,82,87,36,10,76,40,6,7,93,99,39,41,28,86,65,25,63,14,12,43,32,94,24,31,44 1.创建进程 2.结束进程 3.查看内存 4.退出程序 请输入你要进行的操作3 总内存量：100，已使用内存量：80，剩余内存量：20，进程数：2 已使用的内存块为：56,47,1,16,0,73,45,4,9,88,83,2,80,48,30,53,46,68,50,19,34,85,42,98,27,38,18,17,23,11,33,35,8,72,77,67,71,54,96,64,62,69,84,58,26,59,75,13,5,70,81,97,49,15,78,82,87,36,10,76,40,6,7,93,99,39,41,28,86,65,25,63,14,12,43,32,94,24,31,44</pre></div> <div data-bbox="504 1769 1428 2058" data-label="Code-Block"><pre>1.创建进程 2.结束进程 3.查看内存 4.退出程序 请输入你要进行的操作2 1.通过程序号删除 2.通过名称删除 1 请输入程序号 0 已删除0号程序 总内存量：100，已使用内存量：60，剩余内存量：40，进程数：1 已使用的内存块为：56,47,1,16,0,73,45,4,9,88,83,2,80,48,30,53,46,68,50,19,34,85,42,98,27,38,18,17,23,11,33,35,8,72,77,67,71,54,96,64,62,69,84,58,26,59,75,13,5,70,81,97,49,15,78,82,87,36,10,76</pre></div>
--	--

	<div><div>1.创建进程 2.结束进程 3.查看内存 4.退出程序 请输入你要进行的操作2 1.通过程序号删除 2.通过名称删除 2 请输入程序名 进程2 已删除进程2程序 总内存量：100，已使用内存量：10，剩余内存量：90，进程数：0 已使用的内存块为：56,47,1,16,0,73,45,4,9,88</div><div>1.创建进程 2.结束进程 3.查看内存 4.退出程序 请输入你要进行的操作2 1.通过程序号删除 2.通过名称删除 1 请输入程序号 0 无此程序号程序</div><div>无此程序号程序 1.创建进程 2.结束进程 3.查看内存 4.退出程序 请输入你要进行的操作4 Process finished with exit code 0 </div></div>
结 论 (结 果)	成功完成实验
小 结	通过本次实验，加深了我们小组成员对分页存储管理下进程创建，进程结束时的内存空间的分配与回收的理解。

指导老师评议	成绩评定： 指导教师签名：
--------	----------------------