## 《操作系统》实验报告

学号： 姓名： 上课时间：周三上午

|  |  |
| --- | --- |
| **实验名称：** | 存储管理一动态不等长存储资源分配算法 |
| **实验目的：** | 1）理解动态异长存储分区资源管理，掌握所需数据结构和管理程序，了解各种存储分配算法的优点和缺点。 |
| 实验内容： | 1）分析UNIX最先适应(FF)存储分配算法，即map数据结构、存储分配函数malloc()和存储释放函数mfree()，找出与算法有关的成分。   1. 修改上述与算法有关的成分，使其分别体现BF分配原则和WF分配原则。 |
| **程序框图：**   |  |  |  | | --- | --- | --- | | Main函数 | BF/WF\_malloc | mfree | |  |  |  | | |
| **程序关键代码和解读：**   |  |  | | --- | --- | |  | for循环中的bp->m\_size条件，可以控制到表尾时停止，因为表尾m\_size=0。  找到第一个能分配的存储地址后，向后扫描到表尾，是否有更匹配的空间块。找到更匹配的后更新。对于图中的BF方法来说，就是有没有更小的块；对于WF方法来说，就是有没有更大的块。 | |  | 分配完毕后，若该块长度为0，那么将后面的元素全部前移一位。 | |  | Mfree中，先循环找到第一个首地址大于等于释放首地址的。 | |  | 如果前一块存在且其首地址加上大小等于要释放的存储块的首地址，说明可以与前一块进行合并。执行以下操作：将前一块的大小增加释放块的大小。如果要释放的存储块的末尾与后一块的首地址相邻，说明也可以与后一块进行合并。继续循环合并，直到无法合并为止。 | |  | 如果无法与前一块合并，则判断是否可以与后一块合并。如果要释放的存储块的末尾与后一块的首地址相邻且后一块不为空，则可以与后一块合并。执行以下操作：将后一块的首地址减少释放块的大小。将后一块的大小增加释放块的大小。 | |  | 无法合并，把释放块插入数组中，后面元素后移一位。 | | |
| **实验结果分析与体会：**   |  |  | | --- | --- | |  | 如图，首先输入了起始地址0和总长100。然后选择BF算法进行分配。  第一次分配20，结果为<20, 80>。  第二次分配35，结果为<55, 45>。  第三次分配15，结果为<70, 30>。  第四次释放20开始的35个空间，结果为<20, 35>, <70, 30>。  第五次分配15，有两块空间，第二块更适合，因此分配第二块，结果为<20, 35>, <85, 15>符合预期。  第六次释放0开始的20个空间，前后空闲块应当合并，结果为<0, 55>, <85, 15>符合预期。 | |  | 将以上输入，使用WF算法进行分配。在第五次分配时，就会分配第一块，结果为<35, 20>, <70, 30>符合预期。 | | |
| **思考题：**   1. **如何将表中空闲区按照m\_size递增（递减）排序，已得到分配时取第一个长度能满足的表项？**     （2）如何将算法改为最先适应（FF）、下次适应（NF）算法？  对于FF算法，只需在找到第一个匹配的空间后直接返回即可。只需要注释掉BF/WF\_malloc中，找到第一个匹配空间后面的代码即可。  对于NF算法，见图片。   |  |  | | --- | --- | |  |  |   （3）如何更改mfree（），使得不会重复释放资源？  如图： | |