**深 圳 大 学 实 验 报 告**

**课程名称：­ 计算机系统**

**实验项目名称： LC-3汇编语言试验1 求成绩等级**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术**

**指导教师： 蔡晔**

**报告人： 沈晨玙 学号 2019092121 班级：19计科国际班**

**实验时间： 2020年05月20日 星期三**

**实验报告提交时间： 2020年05月20日 星期三**

**教务部制**

|  |
| --- |
| **实验目的：**  （1）分析和理解试验指定的需解决问题。 （2）利用LC-3的汇编语言代码设计实现相关程序。 （3）通过LC-3仿真器调试和运行相关程序并得到正确的结果 |
| **实验内容：**  1.程序起始地址 x3000  2.在x3200处开始存放16个学生成绩，成绩为0~100的正整数  3.编写程序对16个学生的成绩进行排序，从高至低存放在起始地址x4000处  计算学生获得成绩等级为A的 数量 存放在x4100，获得B等级的数量存放在x4101处。  4.成绩等级计算方法：如果一个学生的成绩在85分以上同时排名前25%则获得A，如果一个学生不能获得A，他的成绩在75分以上同时排名前50%，则可获得B。其它成绩则获得C  5.实验报告要写清楚排序算法的实现过程以及等级的计算方法  **实验要求:**  （1）熟悉和理解LC-3的汇编语言指令格式。  （2）掌握在LC-3仿真平台下汇编语言代码的设计输入和调试过程。  （3）掌握利用汇编语言解决问题的思路，加深对底层硬件的理解。 |
| **实验步骤**   1. **分析题目**   题目要求首先将16个成绩从高到低排序并存放在x4100，然后统计等第，所以程序可以分为两个部分。   1. **分析思路并编写程序**   第一部分：排序  思路：  一共16次遍历全部数据，每一次找出当前轮次中的最大值，存放在  [x4100+N]处，一共两层循环。每次遍历出最大值，要注意删除当前最大值（设置为-1），防止影响下一次查找。  删除最大值过程为遍历成绩表，与最大值相等的数据设置为-1。  高级语言形式：  for (i = 16; i > 0, i--) {  max = a[i];  for (j = 15; j > 0; j++) {  if (a[j] > max)  max = a[j];  }  Delete a[j];  max->[x4100 + N];  }  汇编语言代码及注释：  ADD R5,R5,#15;  ADD R5,R5,#1;外层循环16次  LD R2,SCORE2;指向存成绩空间  OUTLOOP ADD R6,R6,#15;内层循环15次  LD R3,SCORE;指针指向成绩  LDR R0,R3,#0;假设第一位为最大值，存入R0  NOT R0,R0;  ADD R0,R0,#1;R0最大值求补数  LOOP1 ADD R3,R3,#1;指针指向下一个数字  LDR R1,R3,#0;R1存入对比数字  ADD R4,R0,R1;R1>R0，将R0设为新的最大值R1，否则继续  BRp SWAP;将最大值设置为当前查询数  ADD R6,R6,#-1;  BRz GOOUT;内层循环结束，跳出二层循环  BRnzp LOOP1;  SWAP AND R0,R0,#0;  NOT R1,R1;  ADD R1,R1,#1;R1最大值求补数  ADD R0,R0,R1;  ADD R6,R6,#-1;  BRz GOOUT;内层循环结束，跳出二层循环  BRnzp LOOP1;  GOOUT LD R7,SCORE  LOOP2 LDR R1,R7,#0;遍历成绩表，与最大值相等的数据设置为-1。  ADD R1,R1,R0;  BRz SET;  ADD R7,R7,#1;  ADD R1,R1,#0;  BRnp LOOP2;  SET LD R1,MINUS1;删除当前轮最大值（设置为-1）  STR R1,R7,#0;  NOT R0,R0;  ADD R0,R0,#1;R0为当前轮最大值  ADD R2,R2,#1;  STR R0,R2,#0;在[x4100+N]存储最大值  ADD R7,R7,#1;  ADD R5,R5,#-1;  BRp OUTLOOP;开始新一轮遍历  BRZ END;  寄存器作用解释：  R0：最大值存储器 R1：存储当前对比的成绩  R2：向目标存储空间的成绩指针 R3：指向成绩保存空间的成绩指针  R4： R5：外层循环计数器  R6：内层循环计数器 R7：向目标存储空间的成绩指针  第二部分：等级赋值  思路：  由题意可知，A等最多4人，B等最多8人。  第一次循环记录A等人数。遍历前4名成绩（一定是前25%），如果大于85分，则为A等，否则进入下一轮。  第一次循环记录B等人数。遍历非A等第一名到第8名成绩（一定是前50%），如果大于75分，则为B等，否则默认位C等。  汇编语言代码及注释：  A等级查询  ADD R1,R1,#4;遍历1-4名成绩，判断是否为A  LD R2,SCORE2;将指针指向第一名成绩  ADD R2,R2,#1;  LD R6,EIGHTYFIVE;R6存储-85  LOOP LDR R0,R2,#0;R0存储当前查询成绩  ADD R4,R0,R6;检查当前成绩是否大于绝对分数85 分，若大于R5=1，小于R5=0  BRzp OVERABSOLUTE;  AND R5,R5,#0;  BRnzp NEXT2;  OVERABSOLUTE AND R5,R5,#0;  ADD R5,R5,#1;  NEXT2 ADD R5,R5,#-1;R5等于1，条件满足，为A，等于0， 条件不满足，进行下一轮测试  BRn NO;  YES ADD R7,R7,#1;R7记录A等人数，加一  ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP;  BRz END1;  NO ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP;  BRz END1;  END1 STI R7,GRADEA;将R7数据存入x4100  **B等级查询**  LD R2,SCORE2;  ADD R2,R2,#1;  ADD R2,R2,R7;将指针指向第N+1名成绩  NOT R7,R7;  ADD R7,R7,#1;  ADD R7,R7,#8;  AND R1,R1,#0;  ADD R1,R1,R7;遍历N+1-8名成绩，判断是否为B,循环次 数8-n存入R1  AND R6,R6,#0;  LD R6,SEVENTYFIVE;R6存储-75  AND R7,R7,#0;人数统计归零  LOOP0 LDR R0,R2,#0;R0存储当前查询成绩  ADD R4,R0,R6;检查当前成绩是否大于绝对分数75分，若 大于R5=1，小于R5=0  BRzp OVERABSOLUTE1;  AND R5,R5,#0;  BRnzp NO1;  OVERABSOLUTE1 AND R5,R5,#0;  ADD R5,R5,#1;  NEXT4 ADD R5,R5,#-1;R5等于1，条件满足，为B，等于0，条 件不满足，默认为C  BRn NO1;  YES1 ADD R7,R7,#1;R7记录A等人数，加一  ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  NO1 ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  END2 STI R7,GRADEB;将R7数据存入x4101  核心代码与A等级查询几乎相同，主要差别在于红色代码部分。通过8与A等级人数的补数相加得到B等级的最大可能人数。从而确认非A第一名的成绩位置。  寄存器作用解释：  R0：存储当前对比的成绩 R1：循环计数器  R2：指向已排序的成绩空间的指针 R3：  R4：临时数据存放 R5：条件成立标志  R6：-85/-75存储 R7：计数器   1. **测试代码**   测试数据1：（无重复数据）  95，82，45，26，98，55，74，86，83，64，51，1，84，76，54，72    运行程序：结果如下  X4000：排序成绩    X4100：A等三人，B等4人    **结果正确！！**  测试数据2：（多组重复数据）  86，86，86，86，86，75，75，75，75，75，75，75，75，75，75，75    运行程序：结果如下  X4000：排序成绩    X4100：A等三人，B等4人    **结果正确！！** |
| **实验结论：**  **第二段等级查询，本来想一次性完成AB两个等地的查询，但是由于寄存器数的限制，所以分为两次完成。**  **代码截图：**  .ORIG X3000  AND R0,R0,#0;  AND R1,R1,#0;  AND R2,R2,#0;  AND R3,R3,#0;  AND R4,R4,#0;  AND R5,R5,#0;  AND R6,R6,#0;  AND R7,R7,#0;初始化寄存器  ADD R5,R5,#15;  ADD R5,R5,#1;外层循环16次  LD R2,SCORE2;指向存成绩空间  OUTLOOP ADD R6,R6,#15;内层循环15次  LD R3,SCORE;指针指向成绩  LDR R0,R3,#0;假设第一位为最大值，存入R0  NOT R0,R0;  ADD R0,R0,#1;R0最大值求补数  LOOP1 ADD R3,R3,#1;指针指向下一个数字  LDR R1,R3,#0;R1存入对比数字  ADD R4,R0,R1;R1>R0，将R0设为新的最大值R1，否则继续  BRp SWAP;将最大值设置为当前查询数  ADD R6,R6,#-1;  BRz GOOUT;内层循环结束，跳出二层循环  BRnzp LOOP1;  SWAP AND R0,R0,#0;  NOT R1,R1;  ADD R1,R1,#1;R1最大值求补数  ADD R0,R0,R1;  ADD R6,R6,#-1;  BRz GOOUT;内层循环结束，跳出二层循环  BRnzp LOOP1;  GOOUT LD R7,SCORE;遍历成绩表，与最大值相等的数据设置为-1。  LOOP2 LDR R1,R7,#0;  ADD R1,R1,R0;  BRz SET;  ADD R7,R7,#1;  ADD R1,R1,#0;  BRnp LOOP2;  SET LD R1,MINUS1;删除当前轮最大值（设置为-1）  STR R1,R7,#0;  NOT R0,R0;  ADD R0,R0,#1;R0为当前轮最大值  ADD R2,R2,#1;  STR R0,R2,#0;  ADD R7,R7,#1;  ADD R5,R5,#-1;  BRp OUTLOOP;  BRZ END;  END AND R0,R0,#0;  AND R1,R1,#0;  AND R2,R2,#0;  AND R3,R3,#0;  AND R4,R4,#0;  AND R5,R5,#0;  AND R6,R6,#0;  AND R7,R7,#0;初始化寄存器    ADD R1,R1,#4;遍历1-4名成绩，判断是否为A  LD R2,SCORE2;将指针指向第一名成绩  ADD R2,R2,#1;  LD R6,EIGHTYFIVE;R6存储-85  LOOP LDR R0,R2,#0;R0存储当前查询成绩  ADD R4,R0,R6;检查当前成绩是否大于绝对分数85分，若大于R5=1，小于R5=0  BRzp OVERABSOLUTE;  AND R5,R5,#0;  BRnzp NEXT2;  OVERABSOLUTE AND R5,R5,#0;  ADD R5,R5,#1;  NEXT2 ADD R5,R5,#-1;R5等于1，条件满足，为A，等于0，条件不满足，进行下一轮测试  BRn NO;  YES ADD R7,R7,#1;R7记录A等人数，加一  ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP;  BRz END1;  NO ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP;  BRz END1;  END1 STI R7,GRADEA;将R7数据存入x4100  LD R2,SCORE2;  ADD R2,R2,#1;  ADD R2,R2,R7;将指针指向第N+1名成绩  NOT R7,R7;  ADD R7,R7,#1;  ADD R7,R7,#8;  AND R1,R1,#0;  ADD R1,R1,R7;遍历N+1-8名成绩，判断是否为B,循环次数8-n存入R1  AND R6,R6,#0;  LD R6,SEVENTYFIVE;R6存储-75  AND R7,R7,#0;人数统计归零  LOOP0 LDR R0,R2,#0;R0存储当前查询成绩  ADD R4,R0,R6;检查当前成绩是否大于绝对分数75分，若大于R5=1，小于R5=0  BRzp OVERABSOLUTE1;  AND R5,R5,#0;  BRnzp NO1;  OVERABSOLUTE1 AND R5,R5,#0;  ADD R5,R5,#1;  NEXT4 ADD R5,R5,#-1;R5等于1，条件满足，为B，等于0，条件不满足，默认为C  BRn NO1;  YES1 ADD R7,R7,#1;R7记录A等人数，加一  ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  NO1 ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  END2 STI R7,GRADEB;将R7数据存入x4101    HALT;  SCORE .fill x3200;  SCORE2 .FILL X3FFF;  GRADEA .fill x4100;  GRADEB .fill x4101;  EIGHTYFIVE .fill #-85;  SEVENTYFIVE .fill #-75;  MINUS1 .FILL #-1;  .END;  ADD R7,R7,#1;  ADD R1,R1,#0;  BRnp LOOP2;  SET LD R1,MINUS1;删除当前轮最大值（设置为-1）  STR R1,R7,#0;  NOT R0,R0;  ADD R0,R0,#1;R0为当前轮最大值  ADD R2,R2,#1;  STR R0,R2,#0;  ADD R7,R7,#1;  ADD R5,R5,#-1;  BRp OUTLOOP;  BRZ END;  END AND R0,R0,#0;  AND R1,R1,#0;  AND R2,R2,#0;  AND R3,R3,#0;  AND R4,R4,#0;  AND R5,R5,#0;  AND R6,R6,#0;  AND R7,R7,#0;初始化寄存器    ADD R1,R1,#4;遍历1-4名成绩，判断是否为A  LD R2,SCORE2;将指针指向第一名成绩  ADD R2,R2,#1;  LD R6,EIGHTYFIVE;R6存储-85  LOOP LDR R0,R2,#0;R0存储当前查询成绩  ADD R4,R0,R6;检查当前成绩是否大于绝对分数85分， 若大于R5=1，小于R5=0  BRzp OVERABSOLUTE;  AND R5,R5,#0;  BRnzp NEXT2;  OVERABSOLUTE AND R5,R5,#0;  ADD R5,R5,#1;  NEXT2 ADD R5,R5,#-1;R5等于1，条件满足，为A，等于0， 条件不满足，进行下一轮测试  BRn NO;  YES ADD R7,R7,#1;R7记录A等人数，加一  ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP;  BRz END1;  NO ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP;  BRz END1;  END1 STI R7,GRADEA;将R7数据存入x4100  LD R2,SCORE2;  ADD R2,R2,#1;  ADD R2,R2,R7;将指针指向第N+1名成绩  NOT R7,R7;  ADD R7,R7,#1;  ADD R7,R7,#8;  AND R1,R1,#0;  ADD R1,R1,R7;遍历N+1-8名成绩，判断是否为B,循环次数8-n存入R1  AND R6,R6,#0;  LD R6,SEVENTYFIVE;R6存储-75  AND R7,R7,#0;人数统计归零  LOOP0 LDR R0,R2,#0;R0存储当前查询成绩  ADD R4,R0,R6;检查当前成绩是否大于绝对分数75分，若大于R5=1，小于R5=0  BRzp OVERABSOLUTE1;  AND R5,R5,#0;  BRnzp NO1;  OVERABSOLUTE1 AND R5,R5,#0;  ADD R5,R5,#1;  NEXT4 ADD R5,R5,#-1;R5等于1，条件满足，为B，等于0，条件不满足，默认为C  BRn NO1;  YES1 ADD R7,R7,#1;R7记录A等人数，加一  ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  NO1 ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  END2 STI R7,GRADEB;将R7数据存入x4101    HALT;  SCORE .fill x3200;  SCORE2 .FILL X3FFF;  GRADEA .fill x4100;  GRADEB .fill x4101;  EIGHTYFIVE .fill #-85;  SEVENTYFIVE .fill #-75;  MINUS1 .FILL #-1;  .END;  BRz END1;  NO ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP;  BRz END1;  END1 STI R7,GRADEA;将R7数据存入x4100  LD R2,SCORE2;  ADD R2,R2,#1;  ADD R2,R2,R7;将指针指向第N+1名成绩  NOT R7,R7;  ADD R7,R7,#1;  ADD R7,R7,#8;  AND R1,R1,#0;  ADD R1,R1,R7;遍历N+1-8名成绩，判断是否为B,循环次数 8-n存入R1  AND R6,R6,#0;  LD R6,SEVENTYFIVE;R6存储-75  AND R7,R7,#0;人数统计归零  LOOP0 LDR R0,R2,#0;R0存储当前查询成绩  ADD R4,R0,R6;检查当前成绩是否大于绝对分数75分，若 大于R5=1，小于R5=0  BRzp OVERABSOLUTE1;  AND R5,R5,#0;  BRnzp NO1;  OVERABSOLUTE1 AND R5,R5,#0;  ADD R5,R5,#1;  NEXT4 ADD R5,R5,#-1;R5等于1，条件满足，为B，等于0，条件 不满足，默认为C  BRn NO1;  YES1 ADD R7,R7,#1;R7记录A等人数，加一  ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  NO1 ADD R2,R2,#1;指针指向下一位成绩  ADD R1,R1,#-1;循环次数减一  BRp LOOP0;  BRz END2;  END2 STI R7,GRADEB;将R7数据存入x4101  HALT;  SCORE .fill x3200;  SCORE2 .FILL X3FFF;  GRADEA .fill x4100;  GRADEB .fill x4101;  EIGHTYFIVE .fill #-85;  SEVENTYFIVE .fill #-75;  MINUS1 .FILL #-1;  .END;  SCORE .fill x3200;  SCORE2 .FILL X3FFF;  GRADEA .fill x4100;  GRADEB .fill x4101;  EIGHTYFIVE .fill #-85;  SEVENTYFIVE .fill #-75;  MINUS1 .FILL #-1;  .END; |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。