**汇编语言程序设计课程实验报告**

**实验8**

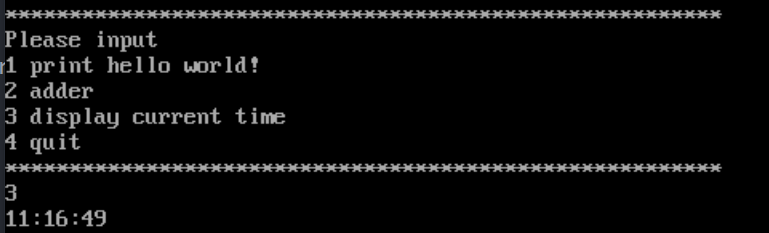
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 姓名 |  | 院系 | 计算机工程与科学学院 | 学号 |  |
| 实验目的： | | | | | |
| 掌握跳跃表法实现多分支程序的方法；掌握子程序设计方法。 | | | | | |
| 实验说明 | | | | | |
| 用户菜单显示操作提示，用户按下菜单键，执行相应的选项。执行完成，重新显示用户菜单，直到用户按下退出键。 | | | | | |
| 实验内容： | | | | | |
| 通过显示功能菜单，实现四个功能。  (1) 在屏幕上显示字符串“hello world!”.(2) 加法器，等待用户输入两个数，并返回加法结果。(3) 显示当前系统时间，以时：分：秒显示.(4) 退出程序. | | | | | |

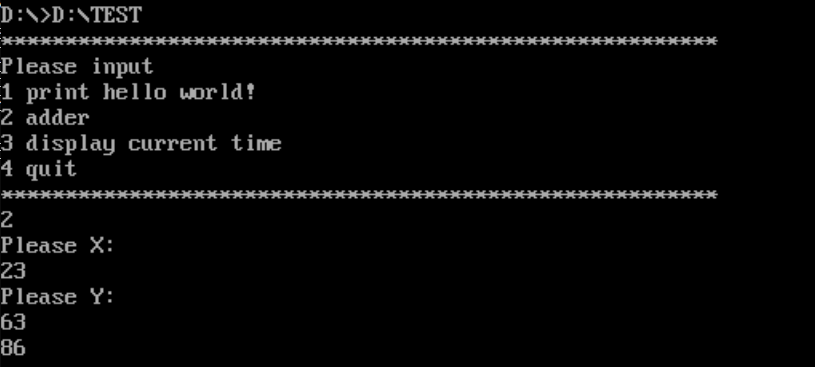
1. 程序分析

程序的基本思路是分为4个“大”子程序进行设计，将子程序的偏移地址存入跳跃表中，由菜单提示用户输入进行选择哪个功能 ，根据用户的输入，找到跳跃表中的位置，进而转入相应的子程序中执行，最后返回菜单栏。

难点在于处理8位数的加法和显示系统时间，前者需要先将用户输入存储为字符串，处理字符串变为二进制数，再将二进制数通过除10操作入栈弹栈进行输出。后者需要调用第2C号dos中断获取系统当时时间，再对时分秒进行二进制转十进制输出，过程较为繁琐。

1. 程序运行结果





1. 程序流程图
2. 实验体会

这次实验的难度适中，功能实现的内容是前几个实验中都有涉及的地方，比如二进制转十进制输出，获取系统时间（实验2内容） ，难点在于理解并使用跳跃表法进行分支结构程序设计，需要搞懂其原理在于jmp的无条件跳转可以改变cs与ip，而且需要注意由于地址是dw类型，故每个子程序地址之间相隔2个地址单位长度。这让我增加了对高级语言中if-elseif-else以及switch-case的分支结构实现的理解。

1. 关键代码

跳跃表跳转与菜单显示

|  |  |
| --- | --- |
|  | memu\_display: |
|  | sub    ax,bX |
|  | sub    bx,bX |
|  | sub    cx,cX |
|  | sub    dx,dX |
|  |  |
|  | lea    dx, menu |
|  | mov    ah,09h |
|  | int    21h |
|  |  |
|  | mov    ah,01h |
|  | int    21h |
|  | call   change |
|  |  |
|  | sub    ah,ah |
|  | sub    al,'0' |
|  | sub    al,1 |
|  | shl    al,1 |
|  |  |
|  | mov    si,ax |
|  | jmp    table[si] |

字符串“十进制”转二进制

|  |  |
| --- | --- |
|  | str2num proc far |
|  | push   bx |
|  | sub    ax,ax |
|  | sub    cx,cx |
|  | sub    dx,dx |
|  |  |
|  | mov    cl,[si] |
|  | mov    bx,10 |
|  | l1: |
|  | add    si,1 |
|  | mov    dl,[si] |
|  | sub    dh,dh |
|  | sub    dl,'0' |
|  | mov    temp,dl |
|  | mul    bx |
|  | add    ax,word ptr temp |
|  | loop   l1 |
|  | pop    bx |
|  | mov    [bx],ax |
|  | ret |
|  | str2num endp |

二进制转十进制输出

|  |  |
| --- | --- |
|  | display\_result proc far |
|  |  |
|  | MOV    AH,00H                 ; |
|  | XOR    CX,CX                  ;      CX记录十进制位数 |
|  | MOV    BL,10                  ;      除数 |
|  | LOOP1: |
|  | DIV    BL                     ;         出发操作，余数在AH，商在AL |
|  | INC    CX                     ;         位数加1 |
|  | PUSH   AX                     ;        入栈保存 |
|  | MOV    AH,00H                 ;     清除余数 |
|  | XOR    AL,00H                 ;     检查是否变为0 |
|  | JNZ    LOOP1                  ;      若还有的除，继续 |
|  |  |
|  | MOV    AH,02H                 ;     AH=02H 输出字符 |
|  | LOOP2: |
|  | POP    DX                     ; |
|  | MOV    DL,DH                  ;      DH里是要输出的余数 |
|  | ADD    DL,30H                 ;     转ASCII码 |
|  | INT    21H                    ;        输出 |
|  | LOOP   LOOP2                  ;     CX = CX-1 JNZ |
|  | ret |
|  | display\_result endp |