

**课 程 实 验 报 告**

**课程名称： 计算机系统基础**

**实验名称： 机器级语言理解和编程**

**院 系 ：计算机科学与技术**

**专业班级 ： 本硕博202001**

**学 号 ： U202115674**

**姓 名 ： 姚晨炫**

**指导教师 ： 许向阳**

**2022 年 10 月 17 日**

**一、实验目的与要求**

(1) 熟练掌握程序开发平台(VS2019) 的基本用法，包括程序的编译、链接和调试；

(2) 熟悉编程的基础知识，包括数据在计算机内的表现形式、寻址方式、常用指令等；

(3) 熟悉程序运行的基本原理；

(4) 熟悉分支、循环程序的结构及控制方法，掌握分支、循环程序的调试方法；

(5) 加深对转移指令及一些常用的汇编指令的理解；

(6) 完成指定功能的程序设计与调试。

**二、实验内容**

**任务4.1 汇编语言程序的编译、链接和调试**

下面汇编语言源程序的功能是：将buf1缓冲区中的12个字节内容拷贝到 buf2中,并显示两个缓冲区中的字符串。

要求使用VS2019进行编译、链接和调试。打开反汇编窗口显示，观察汇编源程序中的语句与反汇编语句之间的关系，分析指令机器码的编码规则。在内存窗口、监视窗口观察数据段的存储结果，说明数据存储规律、各个变量的地址之间的关系等，在寄存器窗口观察寄存器（包括EIP）的值。

汇编语言源程序如下：

.686

.model flat, stdcall

ExitProcess PROTO STDCALL :DWORD

includelib kernel32.lib ; ExitProcess 在 kernel32.lib中实现

printf PROTO C :VARARG

includelib libcmt.lib

includelib legacy\_stdio\_definitions.lib

.DATA

lpFmt db "%s",0ah, 0dh, 0

X DB 10, 255, -1

Y DW 10, 255, -1

Z DD 10, 255, -1

U DW ($-Z)/4

STR1 DB 'Good', 0

P DD X, Y

Q DB 2 DUP (5, 6)

buf1 db 'Hello World',0

buf2 db 12 dup(0) ; 12个字节的空间，初值均为 0

.STACK 200

.CODE

main proc c

MOV ESI,OFFSET buf1

MOV EDI,OFFSET buf2

MOV ECX,0

L1:

MOV EAX, [ESI] ;如果总数不是12个字节，还能每次传送4个字节吗？

MOV [EDI],EAX

ADD ESI, 4

ADD EDI, 4

ADD ECX, 4

CMP ECX,12

JNZ L1

invoke printf,offset lpFmt,OFFSET buf1

invoke printf,offset lpFmt,OFFSET buf2

invoke ExitProcess, 0

main endp

END

**任务4.2 用指定的方式，改写任务4.1中的程序。**

1. 使用变址寻址方式访问 buf1、buf2;
2. 去掉循环，使用直接寻址方式访问 buf1、buf2。

**任务4.3 用汇编语言编写一个输入密码与验证密码的程序。**

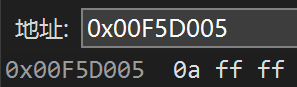
提示输入一个最长10位的字符串密码，与程序内保存的字符串密码比较，相同显示“OK!”, 程序退出；不相同显示“Incorrect Password！”，程序退出。

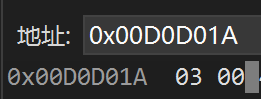
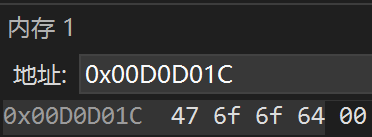
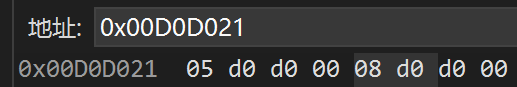
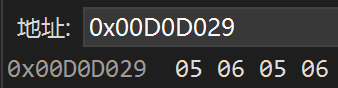
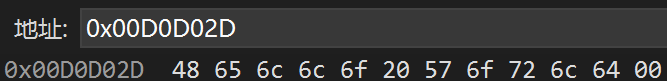
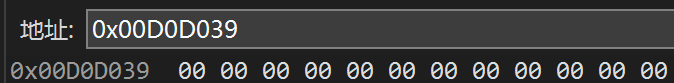
要求：本程序采用分支、循环的程序结构，需要选择合适的字符串比较方法。尝试：采用字节比较、字比较时的程序差异。

**三、实验记录及问题回答**

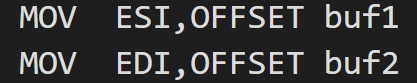
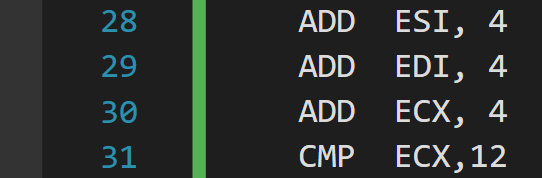
**（1）实验任务 4.1 的实验记录及问题回答**

1.观察.DATA数据段在内存中的存放方式

在实验4.1中X地址下的内存数据为Z地址下内存数据为图形用户界面, 应用程序, 网站

描述已自动生成，其中1在十六进制中表示为0a 255和-1都表示为ff，不同的是，在双字是中，-1的表示为ff ff ff ff，255为 ff 00 00 00。可见在汇编中.DATA段里面，会按照相应的数据存入变量中，更高的位数做相对应的扩展。紧接着是U，，在汇编代码中$代表U现在的地址，($-Z)算的是U和Z的第一个元素存放地址之间的距离，($-Z)/4得到的结果是3。事实上Z每一个元素占四个字节，算的其实是Z的元素个数，为3。U占两个字节，所以为03 00。实验中STR1地址下的内存数据为，对应ascii码的数据为Good 0.对于P，P内存下存放着X和Y的地址，内存中表示为，按照intel机器大端法的读取方式，0x00D0D005为X的地址，0x00D0D008为Y的地址。Q的汇编代码解释为2次存放5，6，对应内存下的数据为5，6，5，6，每一个元素占一个字节，所以为05 06 05 06。。对于buf1，对应的内存存放的数据为，按照ascii码对应过来的数据为Hello World对应的ascii码和数字0，buf2下存放着12个连续的0

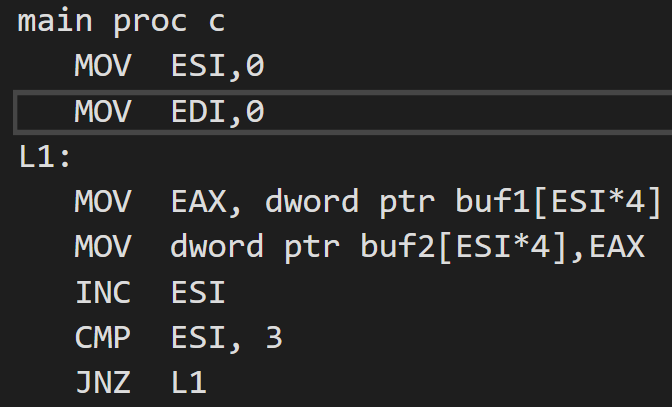
2. 寄存器的相关观察

在这两条汇编指令之后，ESI和EDI下的地址被置为buf1和buf2的地址，，在执行4个add指令之后，对应的寄存器中的数据加了4

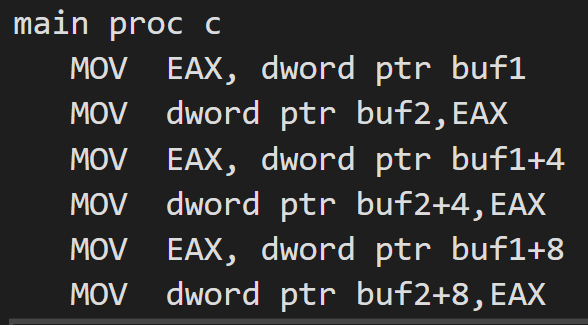
在执行CMP ECX,12语句的时候，EIP寄存器中存放的指令地址为，在执行完jmp指令之后，汇编程序重新开始执行MOV EAX,[ESI]指令，此时EIP中保存的指令地址为，可见程序跳回之前的指令中了。

**（2）实验任务 4.2 的运行结果**

1.使用变址方式访问buf1



2.不用循环，使用直接寻址



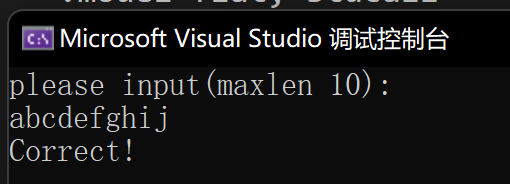
**（3）实验任务 4.3 的算法思想及运行结果**

**1.使用字比较的方式**

**算法思想与流程图**

**图示

描述已自动生成**

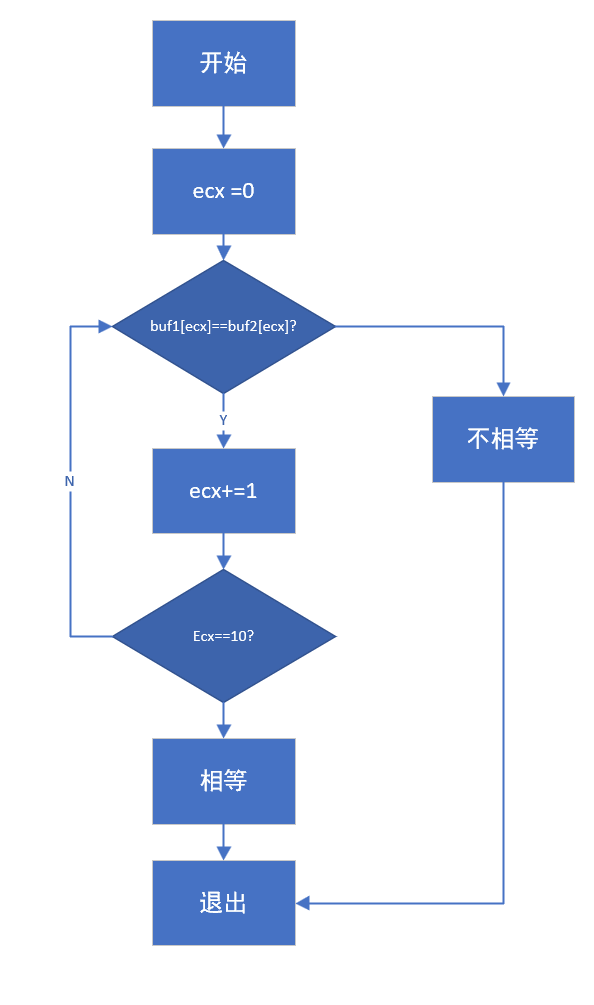


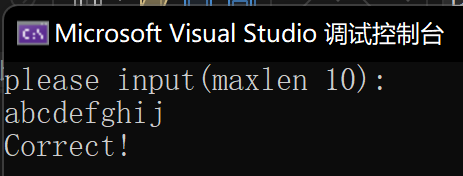
文本

低可信度描述已自动生成

2.使用字节比较的方式

算法思想与流程图





文本

中度可信度描述已自动生成

**四、体会**

实验4.1让我们学习了在汇编代码下的数据的相关存放方式，使用vs2019的反汇编，内存查看等方式观察了数据段的存储结果、存储规律、各个变量的地址之间的关系，练习了不同的寻址方式，让我们对汇编语言的编写和调试更熟悉。4.2让我们使用不同的寻址方式，加深了我们对汇编语言寻址方式的理解。4.3体会到了字节比较和字比较的方法，以及这两种方法的异同。实验三练习了一些运算指令和内存拷贝方法，加深了对数据存储的理解。

**五、源码**

实验任务 4.2 的main 函数的源程序

方式（1）的源代码

方式（2）的源代码

实验任务 4.3 的源程序