

汇编实验一

完成实验后，需用实验报告纸撰写实验报告。

一、实验报告包含以下内容

- 实验序号
- 实验内容
- 算法描述
- 汇编程序
- 运行结果

二、实验目的

1. 掌握汇编器使用，能用汇编器将源代码转变成可执行程序。
2. 熟悉调试器 ollydbg 的使用。
3. 掌握分支结构的汇编表示。

三、指令简介

1 汇编中的乘法指令分为有符号与无符号指令两种。

无符号乘法指令 MUL 指令

- 字节乘法，则 $AL \times REG8/MEM8$ ，乘积存于 AX 中。
- 字乘法，则 $AX \times REG16/MEM16$ ，乘积的高 16 位存放在 DX 中，低 16 位存放在 AX 中。
- 32 位乘法，则 $EAX \times REG32/MEM32$ ，乘积的高 32 位存放在 EDX 中，低 32 位存放在 EAX 中。

2. 无符号除法指令 DIV

- 除数是 REG8/MEM8, 被除数是 16 位寄存器 AX，指令执行完，商存放在 AL，余数存放在 AH。
- 除数是 REG16/MEM16, 被除数是 16 位寄存器 DX 与 16 位寄存器 AX 组成 $(DX) \times 2^{16} + (AX)$ ，指令执行完，商存放在 AX，余数存放在 DX。
- 除数是 REG32/MEM32, 被除数是 32 位寄存器 EDX 与 32 位寄存器 EAX 组成 $(EDX) \times 2^{32} + (EAX)$ ，指令执行完，商存放在 EAX，余数存放在 EDX。

四、控制结构汇编实现方式

1. 对应以下的分支结构(xi,yi 是无符号数)

```
if(x1<y1) {  
    Block_1;  
} else if(x2>y2){  
    Block_2;  
} else if(x3<y3){  
    Block_3;  
}else {  
    Block_4;  
}
```

其汇编实现如下：

```
cmp  x1,y1  
jae  loc2 ;;;<=>jnb  Loc2  
Block_1  
jmp  Final
```

```

Loc2: cmp x2, y2
      jbe Loc3   ;;<=>   jna Loc3
      Block_2
      jmp Final
Loc3: cmp x3,y3
      jae Loc4
      Block_3
      jmp Final
Loc4: Block_4
      jmp Final;(可略)

```

Final:

2. 循环结构的汇编实现

对应 for 循环

```

for(i=0;i<10;i++) {
    BLOCK;
}

```

语义:

```

i=0;
again:  if (i<10){
        BLOCK;
        i=i+1;
        goto again
    }

```

其汇编实现如下

```

      mov  i,0
again:  cmp  i,10
      Jae  final

      BLOCK

      add i,1
      jmp again

```

final:

3. 库函数:

```

Writestring
WriteInt
Writedec
Writechar
Readint

```

五、实验内容

1. 试用汇编语言编写程序，该程序具有如下菜单:

- | | |
|----------|----------|
| [1] 选择排序 | [2] 冒泡排序 |
| [3] 插入排序 | [4] 快速排序 |

- | | |
|-----------|-------------|
| [5] 堆排序 | [6] 全排列 |
| [7] 汉诺塔问题 | [8] 素数环 |
| [9] 整数素分解 | [10] 判定集合元素 |
| [11] 整数拆分 | [12] 八皇后问题 |
| [13] 迷宫问题 | [14] 二分查找 |

请输入选项(1~14):

待输入选项号后，输出所选择的选项功能描述

例如：输入 1，则输出：选择排序

2.实现题 1 中功能[1]即选择排序，对给定的数组排序，然后输出排序后数组；要求用寄存器传递参数

```
int maxIndex(int*arr,int n) {
    int index;
    index=0;
    for(int i=0;i<n;i++) {
        if( arr[index]<arr[i]) {
            index=i
        }
    }
    return index;
}
```

```
void slectSort(int *arr,int n) {
    for(int i=n;i>1;i--){
        j=maxIndex(arr,i);
        int temp=arr[j]
        arr[j]=arr[i-1];
        arr[i-1]=temp;
    }
}
```

测试数据

```
int arr[ ] = {99, 2, 3, -11, 22, 88, 7, 77, 547, 717, -54};
```

3.实现题 1 中功能[3]即插入排序，对给定的数组排序，然后输出排序后数组；要求用栈传递参数。

插入排序的基本操作就是将一个数据插入到已经排好序的有序数据中，从而得到一个新的、个数加一的有序数据，算法适用于少量数据的排序，时间复杂度为 $O(n^2)$ 。是稳定的排序方法。

算法一

```
void insert ( int arr[], int n){
    int key = arr[n];
    int i = n;
    while (arr[i - 1] > key) {
        arr[i] = arr[i-1];
```

```

        i--;
        if(i == 0) break;
    }
    arr[i] = key;
}
void insertionSort( int arr[], int n){
    int i;
    for(i = 1; i < n; i++) {
        insert(arr, i);
    }
}

```

测试数据

```
int arr[ ] = {99, 2, 3, -11, 22, 88, 7, 77, 547, 717, -54};
```

算法二

```

void InsertSortArray ( int arr[], int n)    {
    for (int i = 1; i < n; i++) {
        int temp = arr[i];
        j=i-1;
        while (j >= 0 && arr[j] > temp){
            arr[j+1] = arr[j];
            j--;
        }
        arr[j+1] = temp;
    }
}

```

六、课堂练习

1.试编写汇编程序求无符号整数逆序，比如输入 123456，则输出：654321

2.试求两个整数最大公约数

辗转相除法：

求 x=18 与 y=14 最大公约数

$r_0 = x \% y = 18 \% 14 \quad (=4)$

$r_1 = 14 \% r_0 = 14 \% 4 \quad (=2)$

$r_2 = 4 \% r_1 = 4 \% 2 \quad (=0)$

算法

```
#include <stdio.h>
```

```

int main(){
    int a,b,c;
    printf("请输入两个数:\n");
    scanf("%d%d",&a,&b);
    c=a%b;
    while(c!=0) {

```

```
        a=b;
        b=c;
        c=a%b;
    }
    printf("最大公约数是:\n%d\n",b);
    return 0;
}
```