

西安电子科技大学

微机原理与系统设计 课程实验报告

实验名称 微机原理实验题

人工智能 学院 1920012 班

姓名 杨文韬 学号 18020100245

同作者

实验日期 2021 年 10 月 24 日

成绩

指导教师评语：

指导教师：

年 月 日

实验报告内容基本要求及参考格式

- 一、实验目的
- 二、实验所用仪器（或实验环境）
- 三、实验基本原理及步骤（或方案设计及理论计算）
- 四、实验结果与分析（图表等）
- 五、实验的收获及心得体会
- 六、实验代码

目录

1	实验目的	1
2	实验环境	1
3	方案设计	2
3.1	题目 1 方案	2
3.2	题目 2 方案	2
3.3	题目 3 方案	4
4	实验结果与分析	4
5	实验的收获及心得	5
6	实验代码	5
6.1	题目 1 代码	5
6.2	题目 2 代码	9
6.3	题目 3 代码	15

微机原理实验题

1 实验目的

对于一个编辑好的任汇一语言源程序，会进行译和连接，最终生成一个可执行程序。

(1) 排序：对输入的多个数字进行排序。要求：

- (a) 所有数字从键盘输入；
- (b) 数字中至少包含一个大于 10 的数字；
- (c) 排好序的数字以十进制形式在屏幕显示输出。

(2) 成绩汇总：对输入的一些成绩进行分类汇总。要求：

- (a) 所有数字由键盘输入；
- (b) 输入的成绩个数为任意个 (至少 10 个)；
- (c) 对成绩进行归类并输出显示在屏幕中：
 - 显示最高成绩、最低成绩、平均成绩 (平均成绩保留一位小数)
 - 显示 90-100 分人数，80-89 分人数，70-79 分人数，60-69 人数，低于 60 分人数
 - 显示无效数字个数 (非数字或大于 100 数字个数)

(3) 在字符串中查找自己的学号和姓名，并返回地址

在存储空间定义字符串，该字符串中含有自己的学号和姓名 (拼音)，这两个部分能不能相邻，如：

```
String db "***", "1502031001", "***", "zhang san", "***"
```

要求：在屏幕中显示这两个字符串的偏移地址，并显示学号和姓名。

2 实验环境

- emu8086: 编写汇编语言程序，并进行编译和连接，生成一个可执行程序
- L^AT_EX: 制作封面并进行文档编写排版
- Visio 2016: 程序相关流程图绘制

3 方案设计

3.1 题目 1 方案

题目 1 主体设计流程图如图 1 所示，主体中调用到两个主要的子程序 INPUT 和 BUBBLESORT，分别如图 2 和如图 3 所示，输入的数范围须在 0 ~ 65535(0FFFFH) 之间，采用的排序方法为冒泡排序。

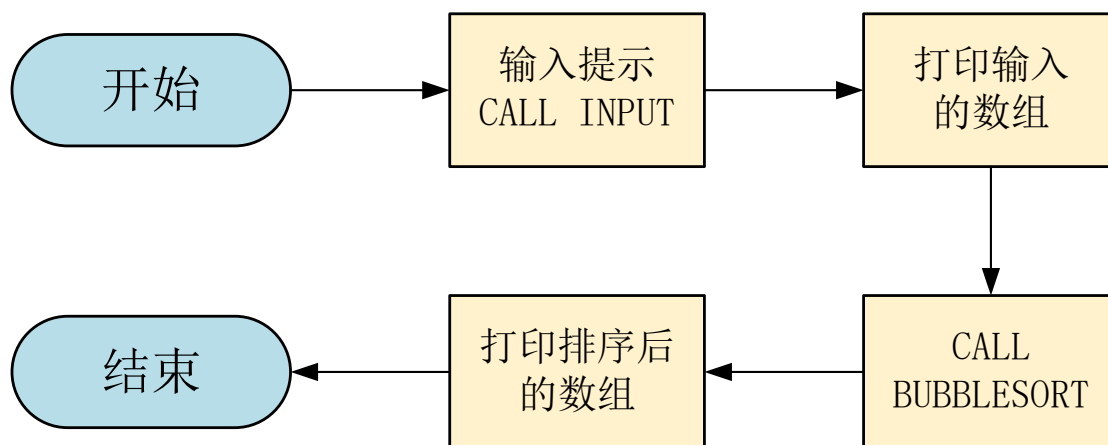


图 1: 题目 1 主体设计流程图

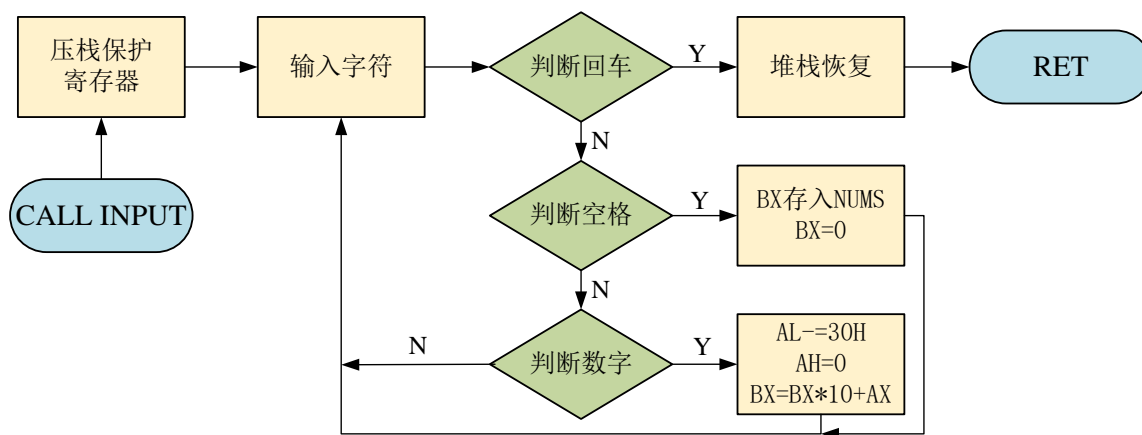


图 2: 题目 1 子程序 INPUT 设计流程图

3.2 题目 2 方案

题目 2 主体设计流程图如图 4 所示。

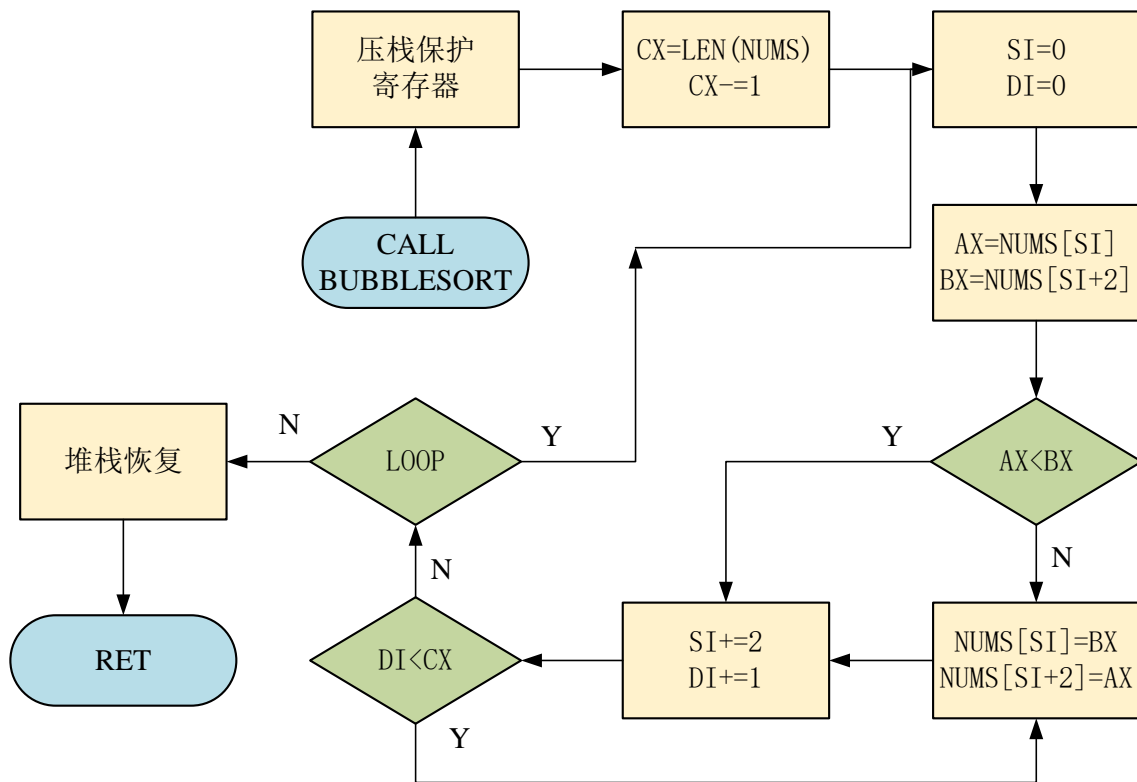


图 3: 题目 1 子程序 BUBBLESORT 设计流程图

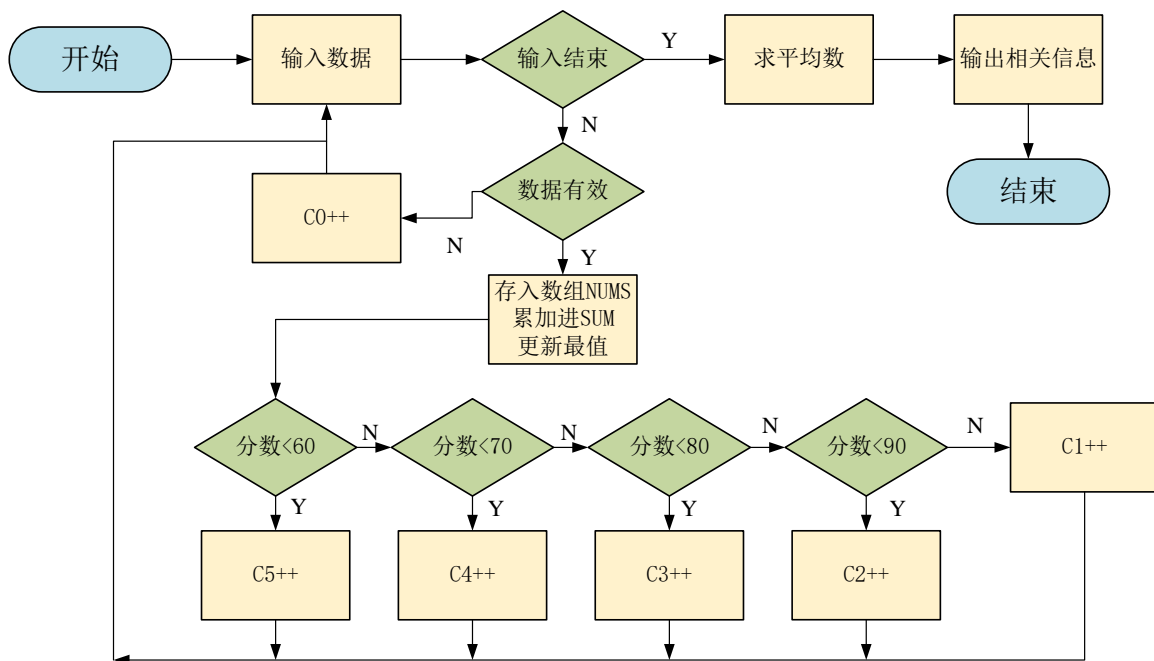


图 4: 题目 2 主体设计流程图

3.3 题目 3 方案

题目 3 主体设计流程图如图 5 所示，题意即在原串中查找模式串。

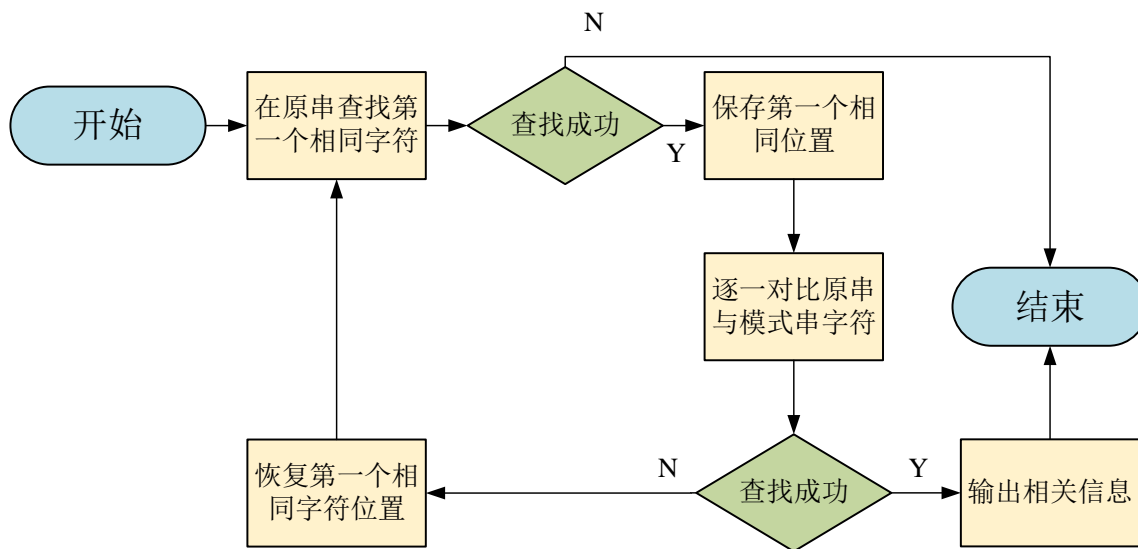


图 5: 题目 3 主体设计流程图

4 实验结果与分析

- 题目 1 输入的数据为 65535 1 99 0 8888 555 321 10000 23 101，其实验结果如图 6 所示。
- 题目 2 输入的数据为 67 88 97 94 78 91 74 68 101 9 asd345 10000 57 73，其中有效数字的平均数为 $\frac{67 + 88 + 97 + 94 + 78 + 91 + 74 + 68 + 9 + 57 + 73}{11} = 72.363636 \approx 72.4$ ，其实验结果如图 7 所示。
- 题目 3 无输入数据，偏移地址从 0 开始，则学号在 3 处找到，姓名在 18 处找到，其实验结果如图 8 所示。

```

PLEASE INPUT NUMBERS WITH SPACE:(0-65535)
65535 1 99 0 8888 555 321 10000 23 101
INPUT NUMBERS ARE:65535 1 99 0 8888 555 321 10000 23 101
SORTED NUMBERS ARE:0 1 23 99 101 321 555 8888 10000 65535
  
```

图 6: 题目 1 实验结果图

```

PLEASE INPUT SCORES WITH SPACE:
67 88 97 94 78 91 74 68 101 9 asd345 10000 57 73
VALID SCORES ARE:67 88 97 94 78 91 74 68 9 57 73

THE HIGHEST SCORE:97
THE LOWEST SCORE:9
THE AVERAGE SCORE:72.4

SCORE BETWEEN 90 AND 100:3
SCORE BETWEEN 80 AND 89:1
SCORE BETWEEN 70 AND 79:3
SCORE BETWEEN 60 AND 69:2
SCORE BELOW 60:2
COUNT OF INVALID INPUT:3

```

图 7: 题目 2 实验结果图

```

STUDENT ID FOUND AT:3
THE PATTERN STRING IS:18020100245
NAME FOUND AT:18
THE PATTERN STRING IS:YANG WEN TAO

```

图 8: 题目 3 实验结果图

5 实验的收获及心得

通过本次实验，我进一步加深了对分支程序设计、循环程序设计和子程序设计的理解，掌握了利用汇编语言实现排序的基本方法和汇编语言中的字符串相关操作，能够较好地将思路转化为流程图，进一步熟悉了 L^AT_EX 排版和 Visio 绘图技巧。有一点感触最深的是，用高级语言实现起来相当容易的算法，用汇编语言实现却会复杂很多，从机器码到汇编语言，从汇编语言到高级语言，是两个非常大的跨越，每一步都大大解放了生产力，学习汇编语言可以让我们更加了解过去的编写程序的方法和加深理解现代计算机的一些概念，对未来的程序设计理解是非常有帮助的。

6 实验代码

6.1 题目 1 代码

```

1 DATA SEGMENT
2     STR1 DB 'PLEASE INPUT NUMBERS WITH SPACE:(0-65535)',13,10,'$'
3     STR2 DB 13,10,'INPUT ERROR! PLEASE INPUT AGAIN:', '$'
4     STR3 DB 13,10,'INPUT NUMBERS ARE:', '$'
5     STR4 DB 13,10,'SORTED NUMBERS ARE:', '$'

```

```

6      NUMS DW 100 DUP(?)
7  DATA ENDS
8
9  STACK SEGMENT STACK 'STACK'
10     DW 256 DUP(?)
11  STACK ENDS
12
13  CODE SEGMENT
14     ASSUME CS:CODE,DS:DATA,SS:STACK
15  START:
16     MOV AX, DATA
17     MOV DS, AX
18     MOV AX, STACK
19     MOV SS, AX
20
21     XOR SI, SI
22     XOR DI, DI
23
24     MOV DX, OFFSET STR1
25     MOV AH, 9
26     INT 21H
27     CALL INPUT
28
29     MOV CX, DI
30     PUSH CX
31
32     MOV DX, OFFSET STR3
33     MOV AH, 9
34     INT 21H
35     XOR DI, DI
36  FOR1:
37     CALL PRINT
38     ADD DI, 2
39     LOOP FOR1
40
41     POP CX
42     CALL BUBBLESORT
43
44     MOV DX, OFFSET STR4
45     MOV AH, 9
46     INT 21H
47     XOR DI, DI
48  FOR2:
49     CALL PRINT
50     ADD DI, 2
51     LOOP FOR2
52

```



```

53     MOV AH, 4CH
54     INT 21H
55
56 ; 输入子程序
57 INPUT PROC NEAR
58     PUSH AX
59     PUSH BX
60     PUSH CX
61     PUSH DX
62
63     XOR BX, BX
64
65 CHECK: ; 输入检查
66     MOV AH, 1
67     INT 21H
68     CMP AL, 0DH ; 回车
69     JZ IN_OVER
70     CMP AL, 20H ; 空格
71     JZ NUMBER
72     SUB AL, 30H
73     JL IN_ERROR
74     CMP AL, 09H
75     JG IN_ERROR
76     MOV AH, 0
77     XCHG AX, BX
78     MOV CX, 10
79     MUL CX
80     JC IN_ERROR
81     ADD AX, BX
82     JC IN_ERROR ; 大于 65535 输入错误
83     XCHG AX, BX
84     JMP CHECK
85
86 NUMBER: ; 输入为数字
87     MOV NUMS[SI], BX
88     ADD SI, 2
89     INC DI
90     XOR BX, BX
91     JMP CHECK
92 IN_ERROR: ; 错误提示
93     XOR BX, BX
94     MOV DX, OFFSET STR2
95     MOV AH, 9
96     INT 21H
97     JMP CHECK
98 IN_OVER: ; 输入结束
99     MOV NUMS[SI], BX

```

```

100     INC DI
101     XOR BX, BX
102
103     POP DX
104     POP CX
105     POP BX
106     POP AX
107     RET
108 INPUT ENDP
109
110 ; 排序子程序
111 BUBBLESORT PROC NEAR
112     PUSH AX
113     PUSH BX
114     PUSH CX
115
116     DEC CX
117 L1:
118     XOR SI, SI
119     XOR DI, DI
120 L2:
121     MOV AX, NUMS[SI]
122     MOV BX, NUMS[SI+2]
123     CMP AX, BX
124     JB L3
125     MOV NUMS[SI], BX
126     MOV NUMS[SI+2], AX
127 L3:
128     ADD SI, 2
129     INC DI
130     CMP DI, CX
131     JB L2
132     LOOP L1
133
134     POP CX
135     POP BX
136     POP AX
137     RET
138 BUBBLESORT ENDP
139
140 ; 输出一个数子程序
141 PRINT PROC NEAR
142     PUSH AX
143     PUSH BX
144     PUSH CX
145     PUSH DX
146

```

```

147     MOV AX, NUMS[DI]
148     MOV BX, 10
149     MOV CX, 0
150 INIT:
151     XOR DX, DX
152     DIV BX ; 商 AX 余 DX
153     INC CX
154     PUSH DX
155     CMP AX, 0
156     JNZ INIT
157 OUTPUT:
158     POP DX
159     OR DX, 30H
160     MOV AH, 2
161     INT 21H
162     LOOP OUTPUT
163
164     MOV DL, 20H
165     MOV AH, 2
166     INT 21H
167
168     POP DX
169     POP CX
170     POP BX
171     POP AX
172     RET
173 PRINT ENDP
174
175 CODE ENDS
176 END START

```

6.2 题目 2 代码

```

1 DATA SEGMENT
2     STR1 DB 'PLEASE INPUT SCORES WITH SPACE:',13,10,'$'
3     STR10 DB 13,10,'VALID SCORES ARE:', '$'
4     STR2 DB 13,10,13,10,'THE HIGHEST SCORE:', '$'
5     STR3 DB 13,10,'THE LOWEST SCORE:', '$'
6     STR4 DB 13,10,'THE AVERAGE SCORE:', '$'
7     STR5 DB 13,10,13,10,'SCORE BETWEEN 90 AND 100:', '$'
8     STR6 DB 13,10,'SCORE BETWEEN 80 AND 89:', '$'
9     STR7 DB 13,10,'SCORE BETWEEN 70 AND 79:', '$'
10    STR8 DB 13,10,'SCORE BETWEEN 60 AND 69:', '$'
11    STR9 DB 13,10,'SCORE BELOW 60:', '$'
12    STR0 DB 13,10,'COUNT OF INVALID INPUT:', '$'

```

```

13     NUMS DW 100 DUP(?)
14     MAXS DW 0 ; HIGHEST SCORE
15     MINS DW 100 ; LOWEST SCORE
16     SUM DW 0 ; SUM OF SCORES
17     C1 DW 0 ; 90-100
18     C2 DW 0 ; 80-89
19     C3 DW 0 ; 70-79
20     C4 DW 0 ; 60-69
21     C5 DW 0 ; 0-59
22     C0 DW 0 ; INVALID
23 DATA ENDS
24
25 STACK SEGMENT STACK 'STACK'
26     DW 256 DUP(?)
27 STACK ENDS
28
29 CODE SEGMENT
30     ASSUME CS:CODE,DS:DATA,SS:STACK
31 START:
32     MOV AX, DATA
33     MOV DS, AX
34     MOV AX, STACK
35     MOV SS, AX
36
37     XOR SI, SI
38     XOR DI, DI
39
40     MOV DX, OFFSET STR1
41     MOV AH, 9
42     INT 21H
43     CALL INPUT
44
45     MOV CX, DI
46     PUSH CX
47
48     MOV DX, OFFSET STR10 ; VALID SCORES
49     MOV AH, 9
50     INT 21H
51     XOR DI, DI
52 FOR1:
53     LEA SI, NUMS[DI]
54     CALL PRINT
55     MOV DL, 20H
56     MOV AH, 2
57     INT 21H
58     ADD DI, 2
59     LOOP FOR1

```

```

60     POP CX
61
62     MOV DX, OFFSET STR2 ; 最高分
63     MOV AH, 9
64     INT 21H
65     LEA SI, MAXS
66     CALL PRINT
67
68     MOV DX, OFFSET STR3 ; 最低分
69     MOV AH, 9
70     INT 21H
71     LEA SI, MINS
72     CALL PRINT
73
74     MOV DX, OFFSET STR4 ; 平均分
75     MOV AH, 9
76     INT 21H
77     XOR DX, DX
78     MOV AX, SUM
79     DIV CX
80     MOV SUM, AX
81     LEA SI, SUM
82     CALL PRINT ; 整数部分
83
84     MOV BX, 10
85     MOV AX, DX
86     MUL BX
87     DIV CX
88     MOV SUM, AX
89     SHL DX, 1
90     CMP DX, CX
91     JL NOCARRY
92     INC SUM
93 NOCARRY:
94     LEA SI, SUM
95
96     MOV DL, 2EH
97     MOV AH, 2
98     INT 21H
99     CALL PRINT ; 小数部分
100
101     MOV DX, OFFSET STR5 ; 90-100
102     MOV AH, 9
103     INT 21H
104     LEA SI, C1
105     CALL PRINT
106

```

```

107     MOV DX, OFFSET STR6 ; 80-89
108     MOV AH, 9
109     INT 21H
110     LEA SI, C2
111     CALL PRINT
112
113     MOV DX, OFFSET STR7 ; 70-79
114     MOV AH, 9
115     INT 21H
116     LEA SI, C3
117     CALL PRINT
118
119     MOV DX, OFFSET STR8 ; 60-69
120     MOV AH, 9
121     INT 21H
122     LEA SI, C4
123     CALL PRINT
124
125     MOV DX, OFFSET STR9 ; 0-59
126     MOV AH, 9
127     INT 21H
128     LEA SI, C5
129     CALL PRINT
130
131     MOV DX, OFFSET STRO ; INVALID
132     MOV AH, 9
133     INT 21H
134     LEA SI, C0
135     CALL PRINT
136
137     MOV AH, 4CH
138     INT 21H
139
140 ; 输入子程序
141 INPUT PROC NEAR
142     PUSH AX
143     PUSH BX
144     PUSH CX
145     PUSH DX
146
147     XOR BX, BX
148     XOR DX, DX
149
150 CHECK:
151     MOV AH, 1
152     INT 21H
153     CMP AL, 0DH ; 回车

```

```

154     JZ IN_OVER
155     CMP AL, 20H ; 空格
156     JZ CONTENT
157     CMP DL, 0
158     JNZ CHECK ; 已经不合法不继续比较
159     SUB AL, 30H
160     JL IN_ERROR
161     CMP AL, 09H
162     JG IN_ERROR
163     MOV AH, 0
164     XCHG AX, BX
165     MOV CX, 10
166     MUL CX
167     ADD AX, BX
168     CMP AX, 100
169     JG IN_ERROR
170     XCHG AX, BX
171     JMP CHECK
172 CONTENT:
173     CMP DL, 0
174     JNZ CASE0
175     MOV NUMS[SI], BX
176     ADD SI, 2
177     INC DI
178     ADD SUM, BX
179     CMP MAXS, BX
180     JL UPDATE_MAX
181     CMP MINS, BX
182     JG UPDATE_MIN
183 COMPARE:
184     CMP BX, 60
185     JL CASE5
186     CMP BX, 70
187     JL CASE4
188     CMP BX, 80
189     JL CASE3
190     CMP BX, 90
191     JL CASE2
192     JMP CASE1
193 UPDATE_MAX: ; UPDATE HIGHEST SCORE
194     MOV MAXS, BX
195     CMP MINS, BX
196     JG UPDATE_MIN
197     JMP COMPARE
198 UPDATE_MIN: ; UPDATE LOWEST SCORE
199     MOV MINS, BX
200     JMP COMPARE

```

```

201 CASE0: ; INVALID
202     INC C0
203     JMP OVER
204 CASE1: ; 90-100
205     INC C1
206     JMP OVER
207 CASE2: ; 80-89
208     INC C2
209     JMP OVER
210 CASE3: ; 70-79
211     INC C3
212     JMP OVER
213 CASE4: ; 60-69
214     INC C4
215     JMP OVER
216 CASE5: ; 0-59
217     INC C5
218     JMP OVER
219 CONTINUE:
220     XOR BX, BX
221     XOR DL, DL
222     JMP CHECK
223 IN_ERROR:
224     XOR BX, BX
225     INC DL ; DL != 0 存在无效数字
226     JMP CHECK
227 IN_OVER:
228     INC DH
229     CMP DL, 0
230     JNZ OVER
231     MOV NUMS[SI], BX
232     INC DI
233
234     ; 最后一个数相关操作
235     ADD SUM, BX
236     CMP MAXS, BX
237     JL UPDATE_MAX
238     CMP MINS, BX
239     JG UPDATE_MIN
240     JMP COMPARE
241 OVER:
242     CMP DH, 0
243     JZ CONTINUE
244
245     POP DX
246     POP CX
247     POP BX

```



```

248     POP AX
249     RET
250 INPUT ENDP
251
252 PRINT PROC NEAR
253     PUSH AX
254     PUSH BX
255     PUSH CX
256     PUSH DX
257
258     MOV AX, [SI]
259     MOV BX, 10
260     MOV CX, 0
261 INIT:
262     XOR DX, DX
263     DIV BX ; 商 AX 余 DX
264     INC CX
265     PUSH DX
266     CMP AX, 0
267     JNZ INIT
268 OUTPUT:
269     POP DX
270     OR DX, 30H
271     MOV AH, 2
272     INT 21H
273     LOOP OUTPUT
274
275     POP DX
276     POP CX
277     POP BX
278     POP AX
279     RET
280 PRINT ENDP
281
282 CODE ENDS
283 END START

```

6.3 题目 3 代码

```

1 DATA SEGMENT
2     STRING DB '***','18020100245','YANG','YANG WEN TAO','***','$'
3     SID DB '18020100245','$'
4     SNAME DB 'YANG WEN TAO','$'
5     STR1 DB 13,10,'STUDENT ID FOUND AT:','$'
6     STR2 DB 13,10,'NAME FOUND AT:','$'

```

```

7      STR DB 13,10,'THE PATTERN STRING IS:','$'
8      LEN DW 33 ; 原串长度
9      LEN1 DW 11 ; 学号长度
10     LEN2 DW 12 ; 姓名长度
11     ADDR DW 1
12 DATA ENDS
13
14 STACK SEGMENT STACK 'STACK'
15     DW 64 DUP(?)
16 STACK ENDS
17
18 CODE SEGMENT
19     ASSUME CS:CODE,DS:DATA,ES:DATA,SS:STACK
20 START:
21     MOV AX, DATA
22     MOV DS, AX
23     MOV ES, AX
24     MOV AX, STACK
25     MOV SS, AX
26
27
28     MOV DX, OFFSET STR1 ; SID
29     MOV AH, 9
30     INT 21H
31     ; 求串长度
32     ; LEA SI, STRING
33     ; CALL STRLEN
34     ; MOV LEN, DI
35
36     LEA SI, SID
37     MOV DX, LEN1 ; DX 存储模式串长度
38
39     CALL SEARCH
40     LEA SI, ADDR
41     CALL PRINT
42
43     MOV DX, OFFSET STR ; 打印模式串
44     MOV AH, 9
45     INT 21H
46
47     LEA SI, STRING
48     ADD SI, ADDR
49     MOV CX, LEN1
50     CALL STRPRT
51
52     MOV DX, OFFSET STR2 ; SNAME
53     MOV AH, 9

```

```

54     INT 21H
55
56     LEA SI, SNAME
57     MOV DX, LEN2
58
59     CALL SEARCH
60     LEA SI, ADDR
61     CALL PRINT
62
63     MOV DX, OFFSET STR ; 打印模式串
64     MOV AH, 9
65     INT 21H
66
67     LEA SI, STRING
68     ADD SI, ADDR
69     MOV CX, LEN2
70     CALL STRPRT
71
72     MOV AH, 4CH
73     INT 21H
74
75 SEARCH PROC NEAR:
76     PUSH AX
77     PUSH CX
78     PUSH DX
79
80     LEA DI, STRING
81     MOV CX, LEN
82     DEC DX
83     MOV AL, [SI]
84
85     CLD
86 COMPARE:
87     REPNZ SCASB ; 找到第一个相同字符
88     JNZ NOTFOUND
89     PUSH SI
90     INC SI
91     PUSH CX
92     PUSH DI ; 保存第一个相同的位置+1
93     MOV CX, DX
94     REPZ CMPSB
95     POP DI
96     POP CX
97     POP SI
98     JNZ COMPARE
99 FOUND:
100    SUB DI, 1

```

```

101     MOV ADDR, DI
102 NOTFOUND:
103
104     POP DX
105     POP CX
106     POP AX
107     RET
108 SEARCH ENDP
109
110 STRLEN PROC NEAR
111     PUSH AX
112     PUSH CX
113
114     XOR CX, CX
115 COUNT:
116     MOV AL, [SI]
117     CMP AL, '$'
118     JZ CNTEND
119     INC CX
120     INC SI
121     JMP COUNT
122 CNTEND:
123     MOV DI, CX
124
125     POP CX
126     POP AX
127     RET
128 STRLEN ENDP
129
130 STRPRT PROC NEAR ; 在原串中显示模式串
131     PUSH AX
132     PUSH CX
133
134 PRT:
135     MOV DL, [SI]
136     MOV AH, 2
137     INT 21H
138     INC SI
139     LOOP PRT
140
141     POP CX
142     POP AX
143     RET
144 STRPRT ENDP
145
146 PRINT PROC NEAR
147     PUSH AX

```

```
148     PUSH BX
149     PUSH CX
150     PUSH DX
151
152     MOV AX, [SI]
153     MOV BX, 10
154     MOV CX, 0
155 INIT:
156     XOR DX, DX
157     DIV BX ; 商 AX 余 DX
158     INC CX
159     PUSH DX
160     CMP AX, 0
161     JNZ INIT
162 OUTPUT:
163     POP DX
164     OR DX, 30H
165     MOV AH, 2
166     INT 21H
167     LOOP OUTPUT
168
169     POP DX
170     POP CX
171     POP BX
172     POP AX
173     RET
174 PRINT ENDP
175
176 CODE ENDS
177 END START
```