

第 3 次上机(x86)

班级	学号	姓名
计试 2201	2223312202	林圣翔

1、子程序设计

(1) 反汇编的截图

TODO: 你的截图

```

0778:0000 B87607      MOV     AX,0776
0778:0003 8ED8        MOV     DS,AX
0778:0005 B90A00      MOV     CX,000A
0778:0008 E80500      CALL    0010
0778:000B B8004C      MOV     AX,4C00
0778:000E CD21        INT     21
0778:0010 BE0000      MOV     SI,0000
0778:0013 B80000      MOV     AX,0000
0778:0016 BA840A00  MOV     AL,[SI+000A]
0778:001A BB0A00      MOV     BX,000A
0778:001D F6F3        DIV     BL
0778:001F 8AD8        MOV     BL,AL
0778:0021 83EB06      SUB     BX,+06
0778:0024 FE871400  INC     BYTE PTR [BX+0014]
0778:0028 83C601      ADD     SI,+01
0778:002B E2E6        LOOP    0013
0778:002D C3          RET
0778:002E 0839      OR      [BX+DI],BH

```

(2) 在进行计算前，显示 ID、array 以及 S6——S10 的内存值的截图（多显示、少显示均扣分）

TODO: 你的截图

```

-d 0 18
0776:0000 32 32 32 33 33 31 32 32-30 32 4C 45 54 5A 49 58 2223312202LETZIX
0776:0010 63 3F 64 50 00 00 00 00-00 c?dP.....

```

(3) 执行完计算后，显示 ID、array 以及 S6——S10 的内存值的截图（多显示、少显示均扣分）

TODO: 你的截图

```

-d 0 18
0776:0000 32 32 32 33 33 31 32 32-30 32 4C 45 54 5A 49 58 2223312202LETZIX
0776:0010 63 3F 64 50 02 02 03 02-01 c?dP.....

```

(4) 源代码

TODO: 你的源代码

```
1  ; empty asm file
2
3  title I love asm
4
5  data segment
6      ID db '2223312202'
7      array db 76,69,84,90,73,88,99,63,100,80
8      s6 db 0
9      s7 db 0
10     s8 db 0
11     s9 db 0
12     s10 db 0
13 data ends
14
15 code segment
16     assume cs:code, ds:data
17     main    proc far
18         ; assign the data segment base address to DS
19         mov  ax, data
20         mov  ds, ax
21         mov  cx,10
22         call calc
23         mov  ax, 4c00h
24         int  21h
25     main    endp
26     calc    proc near
27         mov  si, 0
28     lop:    mov  ax, 0
29
30         mov  al, array[si]
31         mov  bx, 10
32         div  bl
33         mov  bl, al
34         sub  bx, 6
35         inc  s6[bx]
36         add  si, 1
37         loop lop
38         ret
39     calc    endp
40 code ends
41 end main
```

2、高级汇编语言技术

(1) 场景 1 的 .lst 文件的截图

TODO: 你的截图

```
1  FF BS Microsoft (R) Macro Assembler Version 5.00 4/23/24 19:22:02
2  I love asm Page 1-1
3
4
5      1 ; empty asm file
6      2 title I love asm
7      3 0000 data segment
8      4 0000 32 32 32 33 33 31 32 ID db '2223312202'
9      5 32 30 32
10     6 000A 31 32 32 30 32 X db '12202'
11     7 ;X db '2223312202'
12     8 = 0005 len equ $-X
13     9 000F data ends
14
15     10
16     11 0000 code segment
17     12 assume cs:code, ds:data
18     13 0000 main proc
19     14 ; assign the data segment base
20     15 0000 B8 ---- R address to DS mov ax, data
21     16 0003 8E D8 mov ds, ax
22     17 else
23     18 rept len
24     19 add ax, ax
25     20 endm
26     21 0005 03 C0 1 add ax, ax
27     22 0007 03 C0 1 add ax, ax
28     23 0009 03 C0 1 add ax, ax
29
30     24 000B 03 C0 1 add ax, ax
31     25 000D 03 C0 1 add ax, ax
32     26 endif
33     27 000F B8 4C00 mov ax, 4c00h
34     28 0012 CD 21 int 21h
35     29 0014 main endp
36     30 0014 code ends
37     31 end main
38  FF BS Microsoft (R) Macro Assembler Version 5.00 4/23/24 19:22:02
39  I love asm Symbols-1
40
41  Segments and Groups:
42
43      Name Length Align Combine Class
44
45  CODE . . . . . 0014 PARA NONE
46  DATA . . . . . 000F PARA NONE
47
48  Symbols:
49
50      Name Type Value Attr
51
52  ID . . . . . L BYTE 0000 DATA
53
54  LEN . . . . . NUMBER 0005
55
```

```
56  MAIN . . . . . N PROC 0000 CODE Length = 0014
57
58  X . . . . . L BYTE 000A DATA
59
60  @FILENAME . . . . . TEXT d3t2
61
62
63      29 Source Lines
64      34 Total Lines
65      8 Symbols
66
67      50544 + 465904 Bytes symbol space free
68
69      0 Warning Errors
70      0 Severe Errors
71
```

(2) 场景 1 的反汇编的截图

TODO: 你的截图

```
-u 0 14
0777:0000 B87607      MOV     AX,0776
0777:0003 8ED8          MOV     DS,AX
0777:0005 03C0          ADD     AX,AX
0777:0007 03C0          ADD     AX,AX
0777:0009 03C0          ADD     AX,AX
0777:000B 03C0          ADD     AX,AX
0777:000D 03C0          ADD     AX,AX
0777:000F B8004C        MOV     AX,4C00
0777:0012 CD21          INT     21
0777:0014 FF00          INC     WORD PTR [BX+SI]
```

(3) 场景 1 的显示 X 的内存值的截图（多显示、少显示均扣分）

TODO: 你的截图

```
-d a e
0776:0000          31 32 32 30 32          12202
```

(4) 场景 2 的 .lst 文件的截图

TODO: 你的截图

```
1  FF BS Microsoft (R) Macro Assembler Version 5.00 4/23/24 19:26:45
2  I love asm Page 1-1
3
4
5      1          ; empty asm file
6      2          title I love asm
7      3 0000          data segment
8      4 0000 32 32 32 33 33 31 32 ID db '2223312202'
9      5      32 30 32
10     6          ;X db '12202'
11     7 000A 32 32 32 33 33 31 32 X db '2223312202'
12     8      32 30 32
13     9 = 000A          len equ $-X
14    10 0014          data ends
15    11
16    12 0000          code segment
17    13          assume cs:code, ds:data
18    14 0000          main proc
19    15          ; assign the data segment base
20          address to DS
21    16 0000 B8 ---- R          mov ax, data
22    17 0003 8E D8          mov ds, ax
23    18          if len gt 5
24    19          rept 6
25    20          add ax, ax
26    21          endm
27    22 0005 03 C0          1 add ax, ax
28    23 0007 03 C0          1 add ax, ax
```

```
29    24 0009 03 C0          1 add ax, ax
30    25 000B 03 C0          1 add ax, ax
31    26 000D 03 C0          1 add ax, ax
32    27 000F 03 C0          1 add ax, ax
33    28          endif
34    29 0011 B8 4C00          mov ax, 4c00h
35    30 0014 CD 21          int 21h
36    31 0016          main endp
37    32 0016          code ends
38    33          end main
39  FF BS Microsoft (R) Macro Assembler Version 5.00 4/23/24 19:26:45
40  I love asm Symbols-1
41
42
43 Segments and Groups:
44
45      Name Length Align Combine Class
46
47 CODE . . . . . 0016 PARA NONE
48 DATA . . . . . 0014 PARA NONE
49
50 Symbols:
51
52      Name Type Value Attr
53
54 ID . . . . . L BYTE 0000 DATA
```

```
55
56   LEN . . . . . NUMBER 000A
57
58   MAIN . . . . . N PROC 0000 CODE Length = 0016
59
60   X . . . . . L BYTE 000A DATA
61
62   @FILENAME . . . . . TEXT d3t2
63
64
65   | 29 Source Lines
66   | 35 Total Lines
67   | 8 Symbols
68
69   | 50544 + 465904 Bytes symbol space free
70
71   | 0 Warning Errors
72   | 0 Severe Errors
73
```

(5) 场景 2 的反汇编的截图

TODO: 你的截图

```
-u 0 16
0778:0000 B87607      MOV     AX,0776
0778:0003 8ED8          MOV     DS,AX
0778:0005 03C0          ADD     AX,AX
0778:0007 03C0          ADD     AX,AX
0778:0009 03C0          ADD     AX,AX
0778:000B 03C0          ADD     AX,AX
0778:000D 03C0          ADD     AX,AX
0778:000F 03C0          ADD     AX,AX
0778:0011 B8004C          MOV     AX,4C00
0778:0014 CD21          INT     21
0778:0016 83C404          ADD     SP,+04
```

(6) 场景 2 的显示 X 的内存值的截图（多显示、少显示均扣分）

TODO: 你的截图

```
-d a 13
0776:0000                                     32 32 32 33 33 31      222331
0776:0010  32 32 30 32                                     2202
```

(7) 源代码

TODO: 你的源代码

场景一


```

1  ; empty asm file
2  title I love asm
3  data segment
4      ID db '2223312202'
5      X  db '12202'
6      ;X db '2223312202'
7      len equ $-X
8  data ends
9  code segment
10     assume cs:code, ds:data
11     main proc
12         ; assign the data segment base address to DS
13         mov ax, data
14         mov ds, ax
15         if len gt 5
16             rept 6
17                 add ax, ax
18             endm
19         else
20             rept len
21                 add ax, ax
22             endm
23         endif
24         mov ax, 4c00h
25         int 21h
26     main endp
27 code ends
28 end main

```

场景二

```

1  ; empty asm file
2  title I love asm
3  data segment
4      ID db '2223312202'
5      ;X db '12202'
6      X  db '2223312202'
7      len equ $-X
8  data ends
9  code segment
10     assume cs:code, ds:data
11     main proc
12         ; assign the data segment base address to DS
13         mov ax, data
14         mov ds, ax
15         if len gt 5
16             rept 6
17                 add ax, ax
18             endm
19         else
20             rept len
21                 add ax, ax
22             endm
23         endif
24         mov ax, 4c00h
25         int 21h
26     main endp
27 code ends
28 end main

```