* 计科180235 20188068 孔天欣 -

**《汇编语言程序设计》大作业（2019-2020年度）**

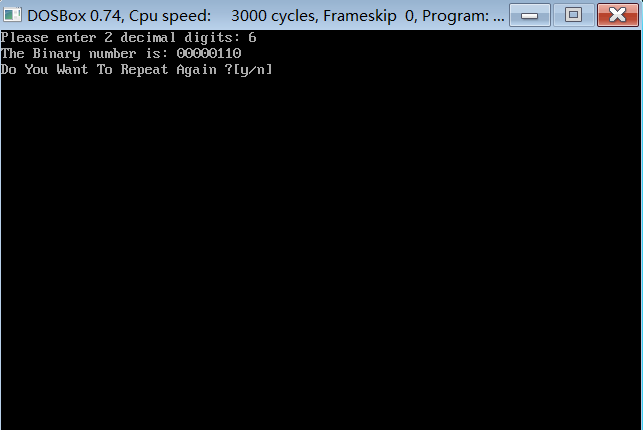
1. 编写程序。从键盘输入一个两位十进制数，输出其二进制结果。

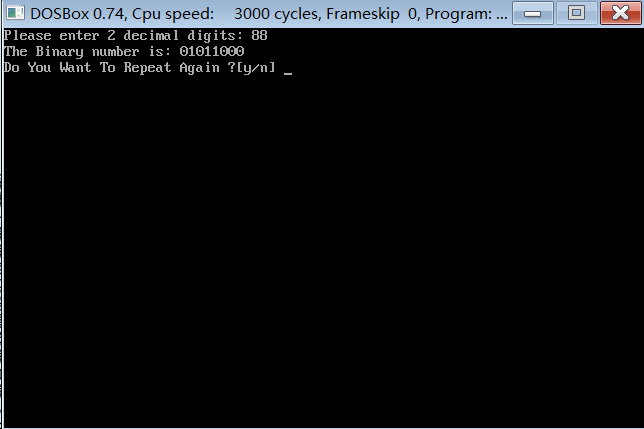
**代码：**

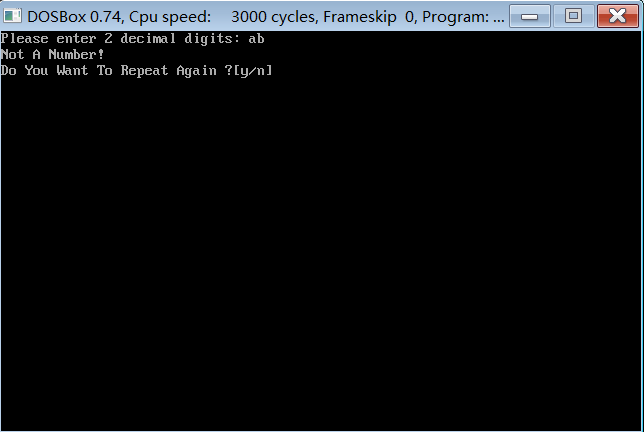
1. ;---------- 题 1 ---------------
3. ;编写程序。从键盘输入一个两位十进制数，
4. ;输出其二进制结果。
6. ;----------- 示 例 --------------
7. ;输入：5
8. ;输出：00000101
10. ;输入：34
11. ;输出：00100010
13. ;输入:340
14. ;输出:Only Two Digits Are Allowed
16. ;输入:Q1
17. ;输出：Not A Number
19. ;----------------------------
20. DATAS SEGMENT
21. ;字符串形式的十进制数字
22. NUMSTR DB 3 DUP(?)
23. ;防止NUMSTR过度填充
24. BUFF\_USELESS DB 32 DUP(0)
25. WRONG\_INFO DB 'Not A Number!$'
26. WRONG\_INFO\_OVER DB 'Only Two Digits Are Allowed!$'
27. ;字符转数字的变量
28. NUMBER DB 0
29. INPUT\_INFO DB 'Please enter 2 decimal digits: $'
30. OUTPUT\_INFO DB 'The Binary number is: $'
31. TIPS\_INFO DB 'Do You Want To Repeat Again ?[y/n] $'
32. DATAS ENDS
34. STACKS SEGMENT
35. ;存放二进制位数的堆栈
36. DW 16 DUP(?)
37. STACKS ENDS
39. CODES SEGMENT
40. ASSUME CS:CODES,DS:DATAS,SS:STACKS
41. START:
42. ;-----------------------------------MACRO
44. ;宏，输出字符
45. PRINTC MACRO STR
46. PROTECT\_STACK
48. MOV AH,2
49. MOV DL,STR
50. INT 21H
52. RECOVER\_STACK
53. ENDM
55. ;宏，输出字符串
56. PRINTS MACRO STR
57. PROTECT\_STACK
58. PUSH DS
60. MOV AX,SEG STR
61. MOV DS,AX
62. LEA DX,STR
63. MOV AH,9
64. INT 21H
66. POP DS
67. RECOVER\_STACK
68. ENDM
70. ;宏，保护现场
71. PROTECT\_STACK MACRO
72. PUSH AX
73. PUSH BX
74. PUSH CX
75. PUSH DX
76. ENDM
78. ;宏，现场恢复
79. RECOVER\_STACK MACRO
80. POP DX
81. POP CX
82. POP BX
83. POP AX
85. ENDM
87. ;宏，换行
88. NEXTLINE MACRO
89. PRINTC 0AH
90. PRINTC 0DH
91. ENDM
93. ;宏 清屏
94. CLEAR\_SCREEN MACRO
96. PROTECT\_STACK
98. MOV AH,6
99. MOV AL,0
100. MOV BH,07H  ;黑底白字
101. MOV CH,0    ;左上(0,0)
102. MOV CL,0
103. MOV DH,24
104. MOV DL,79   ;右下(24,79)->25 \* 80
105. INT 10H
106. ;移动光标到开头
107. MOV AH,2
108. MOV DH,0
109. MOV DL,0
110. MOV BH,0
111. INT 10H
113. RECOVER\_STACK
115. ENDM
117. ;宏，退出程序
118. RTSYS MACRO
119. MOV AH,4CH
120. INT 21H
122. ENDM
124. ;宏，用于反复执行程序
125. DO\_AGAIN MACRO
126. LOCAL RESTART
128. PRINTS TIPS\_INFO
130. MOV AH,1H
131. INT 21H
133. CMP AL,0DH
134. JE RESTART
135. CMP AL,'y'
136. JE RESTART
137. CMP AL,'Y'
138. JE RESTART
139. RTSYS
141. RESTART:
142. CLEAR\_SCREEN
143. JMP STARTS
145. ENDM

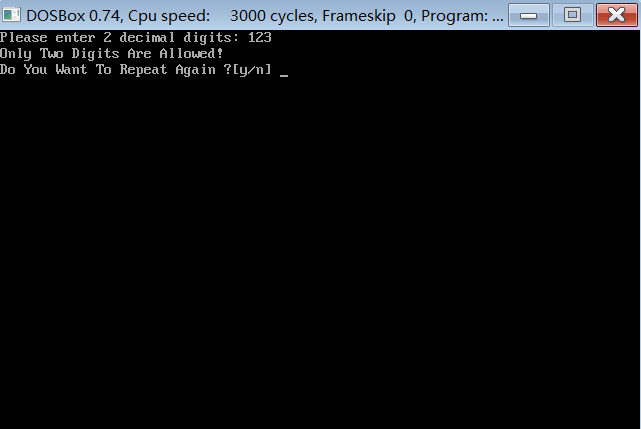

149. ;------------------------------------START
150. STARTS:
151. MOV AX,DATAS
152. MOV DS,AX
153. MOV AX,STACKS
154. MOV SS,AX
156. ;数据重置
157. CALL RESET\_BEGIN
159. ;输出提示消息
160. PRINTS INPUT\_INFO
162. ;输入2位以内十进制数字
163. INPUT:
164. MOV AH,1
165. INT 21H
166. MOV NUMSTR[SI],AL
167. INC SI
168. CMP AL,0DH
169. JNE INPUT
171. ;判断字符是否是数字,如果不是报错
172. CALL JUDGE\_IS\_NUM
174. ;转为十进制数字
175. TOD:
176. ;SI:1->2位
177. MOV SI,1
178. MOV BL,1
179. TODIN:
180. CMP SI,-1
181. JE TOB
182. ;是个位数？
183. MOV AL,NUMSTR[SI]
184. DEC SI
185. CMP AL,0DH
186. JE TODIN
187. ;逐位相乘
188. SUB AL,30H
189. MUL BL
190. ADD NUMBER,AL
191. MOV AL,BL
192. MOV BH,10
193. MUL BH
194. MOV BL,AL
195. JMP TODIN
197. ;转为二进制
198. TOB:
199. MOV AL,NUMBER
200. MOV CX,0
201. TOBIN:
202. MOV BL,2
203. CMP AL,0
204. JE FILL
205. MOV AH,0
206. DIV BL
207. MOV BL,AL
208. MOV AL,0
209. ;余数入栈，商进入下一次循环
210. PUSH AX
211. INC CX
212. MOV AL,BL
213. JMP TOBIN
215. ;8位补足
216. FILL:
217. MOV BX,8
218. SUB BX,CX
219. MOV CX,BX
220. FILLIN:
221. MOV AX,0
222. PUSH AX
223. LOOP FILLIN
225. ;输出提示信息
226. PRINTS OUTPUT\_INFO
228. MOV CX,8
229. ;弹出堆栈，输出各个余数
230. SHOW:
231. POP AX
232. MOV DL,AH
233. ADD DL,30H
234. PRINTC DL
235. LOOP SHOW
236. NEXTLINE
237. JMP LETEND
239. ;非数字，输出错误信息
240. ERROR:
241. PRINTS WRONG\_INFO
242. NEXTLINE
243. JMP LETEND
244. ;字符串长度大于2，报错
245. ERROR\_OVER:
246. PRINTS WRONG\_INFO\_OVER
247. NEXTLINE
249. ;重复程序
250. LETEND:
251. DO\_AGAIN
252. ;---------------------------------------PROC
254. ;判断字符是否是数字,如果不是报错
255. JUDGE\_IS\_NUM PROC
257. PROTECT\_STACK
259. ;只允许输入2位，否则直接报错
260. CMP SI,3
261. JA ERROR\_OVER
263. MOV SI,0
264. MOV CX,2
265. JUDGE:
266. MOV AL,NUMSTR[SI]
267. CMP SI,0
268. JNE JUDGE\_IN
269. ;第一个字符是回车？
270. CMP AL,0DH
271. JE ERROR
272. JUDGE\_IN:
273. INC SI
274. ;是回车？
275. CMP AL,0DH
276. JE TOD
277. ;是0~9的数字？
278. CMP AL,'0'
279. JB ERROR
280. CMP AL,'9'
281. JA ERROR
282. LOOP JUDGE
284. RECOVER\_STACK
286. RET
288. JUDGE\_IS\_NUM ENDP
290. ;数据重置
291. RESET\_BEGIN PROC
292. MOV AX,0
293. MOV SI,0
294. MOV NUMBER,0
295. RET
296. RESET\_BEGIN ENDP
298. CODES ENDS
299. END START

**运行截图：**







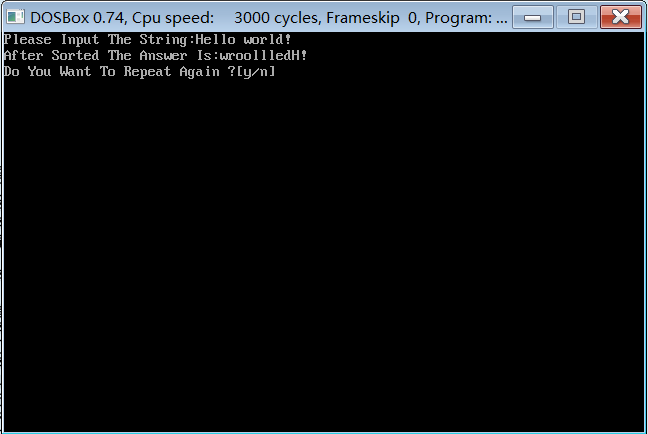


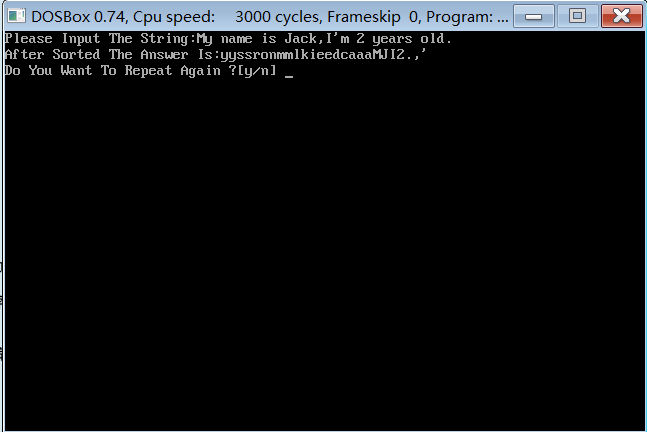
1. 编写程序。从键盘输入一串字符，按照其ASCII码从大到小的顺序排列，并把排序后的结果显示出来。

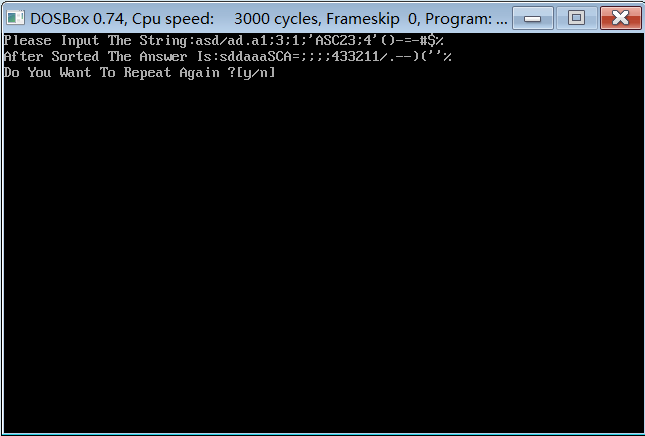
**代码：**

1. ;---------- 题 2 ---------------
3. ;编写程序。从键盘输入一串字符，
4. ;按照其ASCII码从大到小的顺序排列，
5. ;并把排序后的结果显示出来。
7. ;---------- 示 例 --------------
9. ;输入：--==//..123044qwertyuiop
10. ;输出：ywutrqpoie==443210//..--
12. ;----------------------------
13. DATAS SEGMENT
14. INPUTINFO DB 'Please Input The String:$'
15. OUTPUTINFO DB 'After Sorted The Answer Is:$'
16. ;输入的字符串
17. STRING DB 64,?,64 DUP('$')
18. ;NUM=STRING[1]，获取实际输入个数
19. NUM DW 0
20. TIPS\_INFO DB 'Do You Want To Repeat Again ?[y/n] $'
21. DATAS ENDS
23. STACKS SEGMENT
24. DW 32 DUP(?)
25. STACKS ENDS
27. CODES SEGMENT
28. ASSUME CS:CODES,DS:DATAS,SS:STACKS
29. START:
31. ;---------------------------------------MACRO
33. ;宏，输出字符
34. PRINTC MACRO STR
35. PROTECT\_STACK
37. MOV AH,2
38. MOV DL,STR
39. INT 21H
41. RECOVER\_STACK
42. ENDM
44. ;宏，输出字符串
45. PRINTS MACRO STR
46. PROTECT\_STACK
47. PUSH DS
49. MOV AX,SEG STR
50. MOV DS,AX
51. LEA DX,STR
52. MOV AH,9
53. INT 21H
55. POP DS
56. RECOVER\_STACK
57. ENDM
59. ;宏，保护现场
60. PROTECT\_STACK MACRO
61. PUSH AX
62. PUSH BX
63. PUSH CX
64. PUSH DX
65. ENDM
67. ;宏，现场恢复
68. RECOVER\_STACK MACRO
69. POP DX
70. POP CX
71. POP BX
72. POP AX
74. ENDM
76. ;宏，换行
77. NEXTLINE MACRO
78. PRINTC 0AH
79. PRINTC 0DH
80. ENDM
82. ;宏 清屏
83. CLEAR\_SCREEN MACRO
85. PROTECT\_STACK
87. MOV AH,6
88. MOV AL,0
89. MOV BH,07H  ;黑底白字
90. MOV CH,0    ;左上(0,0)
91. MOV CL,0
92. MOV DH,24
93. MOV DL,79   ;右下(24,79)->25 \* 80
94. INT 10H
95. ;移动光标到开头
96. MOV AH,2
97. MOV DH,0
98. MOV DL,0
99. MOV BH,0
100. INT 10H
102. RECOVER\_STACK
104. ENDM
106. ;宏，退出程序
107. RTSYS MACRO
108. MOV AH,4CH
109. INT 21H
111. ENDM
113. ;宏，用于反复执行程序
114. DO\_AGAIN MACRO
115. LOCAL RESTART
117. PRINTS TIPS\_INFO
119. MOV AH,1H
120. INT 21H
122. CMP AL,0DH
123. JE RESTART
124. CMP AL,'y'
125. JE RESTART
126. CMP AL,'Y'
127. JE RESTART
128. RTSYS
130. RESTART:
131. CLEAR\_SCREEN
132. JMP STARTS
134. ENDM
136. ;------------------------------------------START
137. STARTS:
138. MOV AX,DATAS
139. MOV DS,AX
140. MOV AX,STACKS
141. MOV SS,AX
143. ;数据重置
144. CALL RESET\_BEGIN
146. ;输入提示
147. PRINTS INPUTINFO
149. ;输入字符串
150. MOV SI,1
151. LEA DX,STRING
152. MOV AH,10
153. INT 21H
154. ;取字节个数
155. MOV AL,STRING[SI]
156. MOV AH,0
157. MOV NUM,AX
159. ;调用冒泡排序
160. CALL SORT
162. ;回车换行
163. NEXTLINE
164. ;输出提示
165. PRINTS OUTPUTINFO
166. ;输出ASCII排序后的字符串
167. PRINTS STRING[2]
168. NEXTLINE
170. DO\_AGAIN
172. ;------------------------------------------------PROC
174. ;开始冒泡排序
175. SORT PROC
177. CMP NUM,1
178. JE DO\_END
180. ;现场保护
181. PROTECT\_STACK
183. MOV CX,NUM
184. ;CX-1趟
185. DEC CX
186. LOOP1:
187. PUSH CX
188. MOV SI,0
189. LOOP2:
190. MOV AL,STRING[2+SI]
191. CMP AL,STRING[2+SI+1]
192. ;逆序排序
193. JGE NEXT
194. XCHG AL,STRING[2+SI+1]
195. MOV STRING[2+SI],AL
196. NEXT:
197. ADD SI,1
198. LOOP LOOP2
199. POP CX
200. LOOP LOOP1
201. ;恢复现场
202. RECOVER\_STACK
203. DO\_END:
204. RET
205. SORT ENDP
207. ;数据重置
208. RESET\_BEGIN PROC
210. MOV CX,64
211. MOV SI,0
212. RESET:
213. MOV STRING[SI],'$'
214. INC SI
215. LOOP RESET
216. RET
218. RESET\_BEGIN ENDP
220. CODES ENDS
221. END START

**运行截图：**





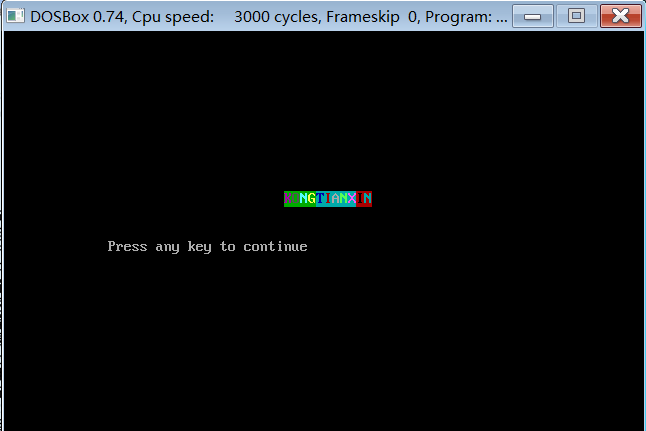


1. 编写90H号软件中断程序，在第（**10**）行（班级序列号**35**）列彩色显示你的**姓名**(注：要求中断驻留，行从第0行开始计算，列从第0列开始计算)。

**代码：**

1. ;---------- 题 3 ---------------
3. ;3. 编写90H号软件中断程序，
4. ;在第（10）行\*\*（班级序列号为35）\*\*列彩色显示你的姓名
5. ;(注：要求中断驻留，行从第0行开始计算，列从第0列开始计算)。
7. ;-----------------------------
9. DATAS SEGMENT
10. DATAS ENDS
12. STACKS SEGMENT
13. STACKS ENDS
15. CODES SEGMENT
16. ASSUME CS:CODES,DS:DATAS,SS:STACKS
17. START:
19. ;-------------------------------------MACRO
21. ;宏，保护现场
22. PROTECT\_STACK MACRO
23. PUSH AX
24. PUSH BX
25. PUSH CX
26. PUSH DX
27. ENDM
29. ;宏，现场恢复
30. RECOVER\_STACK MACRO
31. POP DX
32. POP CX
33. POP BX
34. POP AX
36. ENDM
38. ;宏 清屏
39. CLEAR\_SCREEN MACRO
41. PROTECT\_STACK
43. MOV AH,6
44. MOV AL,0
45. MOV BH,07H  ;黑底白字
46. MOV CH,0    ;左上(0,0)
47. MOV CL,0
48. MOV DH,24
49. MOV DL,79   ;右下(24,79)->25 \* 80
50. INT 10H
51. ;移动光标到开头
52. MOV AH,2
53. MOV DH,0
54. MOV DL,0
55. MOV BH,0
56. INT 10H
58. RECOVER\_STACK
60. ENDM
61. ;---------------------------------------START
63. ;将90H号中断放入向量表中
64. LEA DX,OUTPUTSTART
65. MOV AX,SEG OUTPUTSTART
66. MOV DS,AX
67. MOV AL,90H
68. MOV AH,25H
69. INT 21H
71. ;中断驻留
72. MOV AH,31H
73. MOV AL,0
74. MOV DX,OUTPUTEND-OUTPUTSTART+16
76. ;触发90H号中断
77. INT 90H
79. MOV AH,4CH
80. INT 21H
82. ;中断程序
83. OUTPUTSTART:
85. JMP CODE
86. ;姓名
87. MYNAME DB 'KONGTIANXIN'
88. ;颜色
89. COLOR DB 1EH
90. CODE:
92. ;现场保护
93. PUSH DS
94. PROTECT\_STACK
95. PUSH CS
96. POP DS
98. CLEAR\_SCREEN
100. ;移动光标到10行35列
101. MOV AH,2
102. MOV DH,10
103. MOV DL,35
104. MOV BH,0
105. INT 10H
107. ;写显存
108. MOV BX,0B800H
109. MOV ES,BX
110. ;行：A0H \* AH = 640H
111. MOV BX,640H
112. ;列 2H \* 23H(35D) = 46H
113. MOV DI,46H
114. ;名字的长度作为循环
115. MOV CX,DWORD PTR COLOR-MYNAME
116. MOV SI,0
117. PAINT:
118. ;输出字符
119. MOV AH,2
120. MOV DL,MYNAME[SI]
121. INC SI
122. INT 21H
123. MOV AL,COLOR
124. ;填色，奇地址存放属性
125. ADD BYTE PTR ES:[BX+1+DI],AL
126. ;下一个字符换颜色
127. ADD AL,3H
128. MOV COLOR,AL
129. ;下一个相邻位置
130. ADD BX,2
131. LOOP PAINT
133. ;现场恢复
134. RECOVER\_STACK
135. POP DS
136. IRET
137. OUTPUTEND:NOP
138. CODES ENDS
139. END START

**运行截图：**



1. 编写程序。从键盘输入多个带符号数：

①统计负数的个数；

②计算所有负数之和，并显示出十进制结果。

**代码：**

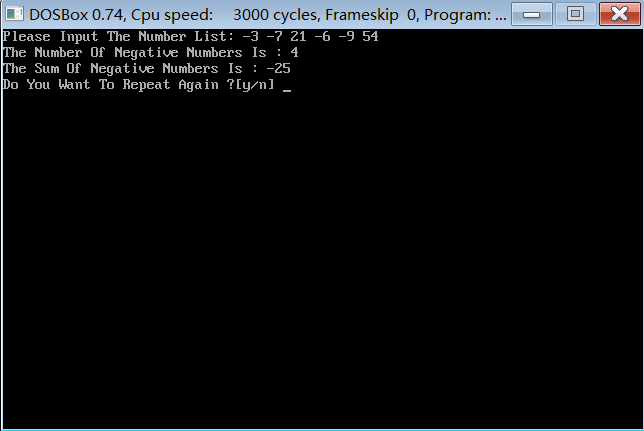
1. ;---------- 题 4 ---------------
3. ;编写程序。从键盘输入多个带符号数：
4. ;①统计负数的个数；
5. ;②计算所有负数之和，并显示出十进制结果。
7. ;---------- 示 例 --------------

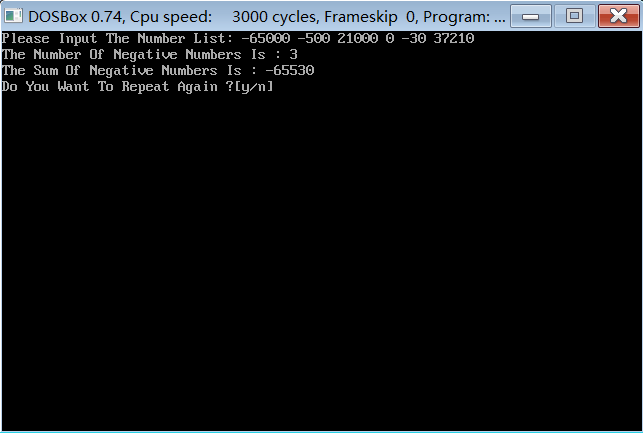
10. ;输入：34 -2 -3 -4 43
11. ;输出：The Number Of Negative Numbers **Is** : 3
12. ;      The Sum Of Negative Numbers **Is** : -9

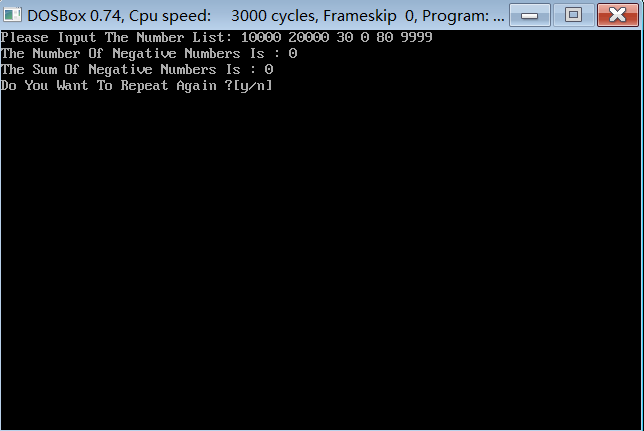
15. ;输入:0 -200 -100 -50000 50000
16. ;输出：The Number Of Negative Numbers **Is** : 3
17. ;      The Sum Of Negative Numbers **Is** : -50300
19. ;输入:0 200 100 50000 50000
20. ;输出：The Number Of Negative Numbers **Is** : 0
21. ;      The Sum Of Negative Numbers **Is** : 0
23. ;输入: 23 33 5G
24. ;输出：Please Input Correct Synatax
26. ;----------------------------
28. DATAS SEGMENT
29. ;存放数字字符串，以$分割
30. ARRAYSTR DB 128 DUP('$')
31. ;ARRAYSTR的实际最大下标
32. PTRNUM DW 0
33. NUM\_OF\_NEGA DB 0
34. ;临时数字，用于暂存十进制数（无论正负）
35. TEMP DW 0
36. ;是否是负数的标志
37. SIGN DB 0
38. ;负数的总和,上限65535
39. SUM DW 0
40. INPUT\_INFO DB 'Please Input The Number List: $'
41. ;错误信息：请输入正确的格式
42. WRONG\_INFO DB 'Please Input Correct Synatax!$'
43. NUM\_INFO DB 'The Number Of Negative Numbers Is : $'
44. SUM\_INFO DB 'The Sum Of Negative Numbers Is : $'
45. OVER\_INFO DB 'Overflow! $'
46. TIPS\_INFO DB 'Do You Want To Repeat Again ?[y/n] $'
47. DATAS ENDS
49. STACKS SEGMENT
50. STACKS ENDS
52. CODES SEGMENT
53. ASSUME CS:CODES,DS:DATAS,SS:STACKS
54. START:
56. ;--------------------------------MACRO
58. ;宏，输出字符
59. PRINTC MACRO STR
60. PROTECT\_STACK
62. MOV AH,2
63. MOV DL,STR
64. INT 21H
66. RECOVER\_STACK
67. ENDM
69. ;宏，输出字符串
70. PRINTS MACRO STR
71. PROTECT\_STACK
72. PUSH DS
74. MOV AX,SEG STR
75. MOV DS,AX
76. LEA DX,STR
77. MOV AH,9
78. INT 21H
80. POP DS
81. RECOVER\_STACK
82. ENDM
84. ;宏，保护现场
85. PROTECT\_STACK MACRO
86. PUSH AX
87. PUSH BX
88. PUSH CX
89. PUSH DX
90. ENDM
92. ;宏，现场恢复
93. RECOVER\_STACK MACRO
94. POP DX
95. POP CX
96. POP BX
97. POP AX
99. ENDM
101. ;宏，换行
102. NEXTLINE MACRO
103. PRINTC 0AH
104. PRINTC 0DH
105. ENDM
107. ;宏 清屏
108. CLEAR\_SCREEN MACRO
110. PROTECT\_STACK
112. MOV AH,6
113. MOV AL,0
114. MOV BH,07H  ;黑底白字
115. MOV CH,0    ;左上(0,0)
116. MOV CL,0
117. MOV DH,24
118. MOV DL,79   ;右下(24,79)->25 \* 80
119. INT 10H
120. ;移动光标到开头
121. MOV AH,2
122. MOV DH,0
123. MOV DL,0
124. MOV BH,0
125. INT 10H
127. RECOVER\_STACK
129. ENDM
131. ;宏，退出程序
132. RTSYS MACRO
133. MOV AH,4CH
134. INT 21H
136. ENDM
138. ;宏，用于反复执行程序
139. DO\_AGAIN MACRO
140. LOCAL RESTART
142. PRINTS TIPS\_INFO
144. MOV AH,1H
145. INT 21H
147. CMP AL,0DH
148. JE RESTART
149. CMP AL,'y'
150. JE RESTART
151. CMP AL,'Y'
152. JE RESTART
153. RTSYS
155. RESTART:
156. CLEAR\_SCREEN
157. JMP STARTS
159. ENDM

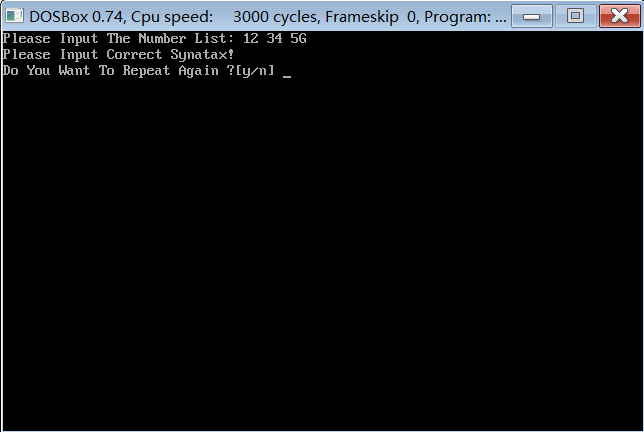
162. ;宏，输出数字的十进制字符串
163. PRINT\_NUM MACRO NUM
165. LOCAL DO\_ING
166. LOCAL DO\_END
168. ;保护现场
169. PROTECT\_STACK
171. MOV AX,WORD PTR NUM
172. MOV DX,0
173. MOV BX,10
174. DO\_ING:
175. DIV BX
176. ;余数入栈，商进入下一次循环
177. PUSH DX
178. MOV DX,0
179. INC CX
180. CMP AX,0
181. JE DO\_END
182. JMP DO\_ING
183. ;按位输出
184. DO\_END:
185. POP AX
186. ADD AX,3030H
187. PRINTC AL
188. LOOP DO\_END
190. ;现场恢复
191. RECOVER\_STACK
193. ENDM
195. ;-----------------------------START
196. STARTS:
197. MOV AX,DATAS
198. MOV DS,AX
200. ;重置数据
201. CALL RESET\_BEGIN
203. PRINTS INPUT\_INFO
205. ;先输入一串带符号数，空格作为分割，
206. ;在存储单元中以'$'代替
207. INPUT:
208. MOV AH,1
209. INT 21H
211. ;是回车？
212. CMP AL,0DH
213. ;跳出循环
214. JE TONUM
216. ;是空格？
217. CMP AL,20H
218. JNE ISNEGA
219. ;分隔符
220. INC SI
221. JMP INPUT
223. ;是负号？
224. ISNEGA:
225. CMP AL,'-'
226. JNE NOTNEGA
227. MOV ARRAYSTR[SI],'-'
228. INC SI
229. JMP INPUT
230. ;不是负号的情况下
231. NOTNEGA:
232. ;是0~9的数字？
233. CMP AL,'0'
234. ;不是，报错
235. JB ERROR
236. CMP AL,'9'
237. JA ERROR
238. MOV ARRAYSTR[SI],AL
239. INC SI
240. JMP INPUT
242. ;所有负数相加，并统计负数个数
243. TONUM:
244. MOV PTRNUM,SI
245. ;数组下标
246. MOV SI,0
247. ;一个数字除符号外的位数
248. MOV CX,0
249. ;乘数
250. MOV BX,1
251. TONUMIN:
253. ;如果数组下标已经大于PTRNUM，跳出
254. CMP PTRNUM,SI
255. JB PRINT\_ALL
257. MOV AL,ARRAYSTR[SI]
258. CMP AL,'$'
259. JE TOSUM
260. CMP AL,'-'
261. JE ADDNEGA
262. ;是数字字符，入栈
263. MOV AH,0
264. PUSH AX
265. INC SI
266. INC CX
267. JMP TONUMIN
268. ADDNEGA:
269. ;是负数
270. MOV SIGN,1
271. INC NUM\_OF\_NEGA
272. INC SI
273. JMP TONUMIN
274. TOSUM:
275. ;从个位数出栈
276. POP AX
277. SUB AL,30H
278. MUL BX
279. ADD TEMP,AX
280. MOV AX,BX
281. MOV BX,10
282. MUL BX
283. MOV BX,AX
284. LOOP TOSUM
285. ;是负数吗？
286. CMP SIGN,1
287. JNE TOSUMIN
288. ;是负数，SUM+=TEMP，重置
289. MOV AX,TEMP
290. ADD SUM,AX
291. INC SI
292. CALL RESET
293. JMP TONUMIN
294. ;不是负数，重置
295. TOSUMIN:
296. INC SI
297. CALL RESET
298. JMP TONUMIN
300. ;输出所有消息
301. PRINT\_ALL:
303. PRINTS NUM\_INFO
304. PRINT\_NUM NUM\_OF\_NEGA
305. NEXTLINE
307. PRINTS SUM\_INFO
309. ;没有负数也就没必要打印符号
310. CMP NUM\_OF\_NEGA,0
311. JE DO\_NOT\_PRINTN
312. PRINTC '-'
313. DO\_NOT\_PRINTN:
314. PRINT\_NUM SUM
315. NEXTLINE
317. JMP LETE
318. ;输出错误消息
319. ERROR:
320. NEXTLINE
321. PRINTS WRONG\_INFO
322. NEXTLINE
323. JMP LETE
325. LETE:
327. DO\_AGAIN
329. ;--------------------------------PROC
331. ;重置相关信息
332. RESET PROC
333. MOV TEMP,0
334. MOV BX,1
335. MOV SIGN,0
336. RET
337. RESET ENDP
339. ;数据重置
340. RESET\_BEGIN PROC
342. MOV SI,0
343. MOV TEMP,0
344. MOV SUM,0
345. MOV NUM\_OF\_NEGA,0
346. MOV CX,128
347. RESET\_CLEAR:
348. MOV ARRAYSTR[SI],'$'
349. INC SI
350. LOOP RESET\_CLEAR
351. MOV SI,0
352. RET
354. RESET\_BEGIN ENDP
356. CODES ENDS
357. END START

**运行截图：**









5．读取系统时间的百分秒作为0—99之间的随机数，实现两个随机数的加、减、乘、除运算，并显示算式及计算结果。

**代码：**

1. ;---------- 题 5 ---------------
3. ;读取系统时间的百分秒作为0—99之间的随机数，
4. ;实现两个随机数的加、减、乘、除运算，
5. ;并显示算式及计算结果。
7. ;-----------------------------
9. DATAS SEGMENT
10. ;MTIME：存放百分秒数值
11. MTIME\_1 DW 0
12. MTIME\_2 DW 0
13. NUM\_INFO\_BEFORE DB 'The Frist Number Is : $'
14. NUM\_INFO\_AFTER DB 'The Second Number Is : $'
15. ADD\_INFO DB 'NUM1 + NUM2 = $'
16. SUB\_INFO DB 'NUM1 - NUM2 = $'
17. MUL\_INFO DB 'NUM1 \* NUM2 = $'
18. DIV\_INFO DB 'NUM1 / NUM2 = $'
19. MOD\_INFO DB ' ,MOD = $'
20. TIPS\_INFO DB 'Do You Want To Repeat Again ?[y/n] $'
21. DELAY\_TIME DB 14
22. DATAS ENDS
24. STACKS SEGMENT
26. STACKS ENDS
28. CODES SEGMENT
29. ASSUME CS:CODES,DS:DATAS,SS:STACKS
30. START:
32. ;-------------------------------------MACRO
34. ;宏，输出字符
35. PRINTC MACRO STR
36. PROTECT\_STACK
38. MOV AH,2
39. MOV DL,STR
40. INT 21H
42. RECOVER\_STACK
43. ENDM
45. ;宏，输出字符串
46. PRINTS MACRO STR
47. PROTECT\_STACK
48. PUSH DS
50. MOV AX,SEG STR
51. MOV DS,AX
52. LEA DX,STR
53. MOV AH,9
54. INT 21H
56. POP DS
57. RECOVER\_STACK
58. ENDM
60. PROTECT\_STACK MACRO
61. PUSH AX
62. PUSH BX
63. PUSH CX
64. PUSH DX
65. ENDM
67. RECOVER\_STACK MACRO
68. POP DX
69. POP CX
70. POP BX
71. POP AX
73. ENDM
75. ;宏，换行
76. NEXTLINE MACRO
77. PRINTC 0AH
78. PRINTC 0DH
79. ENDM
81. ;宏 清屏
82. CLEAR\_SCREEN MACRO
84. PROTECT\_STACK
86. MOV AH,6
87. MOV AL,0
88. MOV BH,07H  ;黑底白字
89. MOV CH,0    ;左上(0,0)
90. MOV CL,0
91. MOV DH,24
92. MOV DL,79   ;右下(24,79)->25 \* 80
93. INT 10H
94. ;移动光标到开头
95. MOV AH,2
96. MOV DH,0
97. MOV DL,0
98. MOV BH,0
99. INT 10H
101. RECOVER\_STACK
103. ENDM
105. ;宏，输出数字的十进制字符串
106. PRINT\_NUMS MACRO NUM
108. LOCAL DO\_ING
109. LOCAL DO\_END
111. ;保护现场
112. PROTECT\_STACK
114. MOV AX,NUM
115. MOV DX,0
116. MOV CX,0
117. MOV BX,10
118. DO\_ING:
119. DIV BX
120. ;余数入栈，商进入下一次循环
121. PUSH DX
122. MOV DX,0
123. INC CX
124. CMP AX,0
125. JE DO\_END
126. JMP DO\_ING
127. ;按位输出
128. DO\_END:
129. POP AX
130. ADD AX,3030H
131. PRINTC AL
132. LOOP DO\_END
134. ;现场恢复
135. RECOVER\_STACK
136. ENDM
138. ;宏，获取当前时间
139. GET\_MTIME MACRO ARG,INFO
141. MOV AH,2CH      ;DL->millsecond
142. INT 21H
143. MOV DH,0
144. MOV ARG,DX
146. PRINTS INFO
147. PRINT\_NUMS ARG
148. NEXTLINE
150. ENDM
152. ;宏，退出程序
153. RTSYS MACRO
154. MOV AH,4CH
155. INT 21H
157. ENDM
159. ;宏，反复执行程序
160. DO\_AGAIN MACRO
161. LOCAL RESTART
163. PRINTS TIPS\_INFO
165. MOV AH,1H
166. INT 21H
168. CMP AL,0DH
169. JE RESTART
170. CMP AL,'y'
171. JE RESTART
172. CMP AL,'Y'
173. JE RESTART
174. RTSYS
176. RESTART:
177. CLEAR\_SCREEN
178. JMP STARTS
179. ENDM
180. ;-------------------------------------------中断
182. ;---除0(0号)的中断改写-----
184. ;将00H号中断放入向量表中
185. LEA DX,DIVIDE\_START
186. MOV AX,SEG DIVIDE\_START
187. MOV DS,AX
188. MOV AL,00H
189. MOV AH,25H
190. INT 21H
192. ;中断驻留
193. MOV AH,31H
194. MOV AL,0
195. MOV DX,DIVIDE\_END-DIVIDE\_START+16
197. JMP STARTS
199. ;中断程序
200. DIVIDE\_START:
201. JMP CODE
202. DIVIDE\_ERROR DB '0 Cannot Be Used As A Divisor.So Please \*RESTART\* The Program.$'
203. CODE:
205. ;现场保护
206. PUSH DS
207. PUSH DX
208. PUSH AX
209. PUSH CS
210. POP DS
212. PRINTS DIVIDE\_ERROR
214. ;现场恢复
215. POP AX
216. POP DX
217. POP DS
219. ;回到界面
220. MOV AX,4C00H
221. INT 21H
223. DIVIDE\_END:NOP
225. ;-------------------------------------------START
226. STARTS:
227. MOV AX,DATAS
228. MOV DS,AX
230. ;MTIME\_1 = 当前百分秒
231. GET\_MTIME MTIME\_1,NUM\_INFO\_BEFORE
233. ;执行延时
234. CALL DELAY
236. ;MTIME\_1 = 延时后的百分秒
237. GET\_MTIME MTIME\_2,NUM\_INFO\_AFTER
239. ;加
240. CALL DO\_ADD
242. ;减
243. CALL DO\_SUB
245. ;乘
246. CALL DO\_MUL
248. ;除
249. CALL DO\_DIV
251. ;反复执行程序
252. DO\_AGAIN

255. ;----------------------------------------PROC
257. ;加法
258. DO\_ADD PROC
260. MOV AX,MTIME\_1
261. MOV BX,MTIME\_2
262. ADD AX,BX
263. ;打印相加结果
264. PRINTS ADD\_INFO
265. PRINT\_NUMS AX
266. NEXTLINE
267. RET
269. DO\_ADD ENDP

272. ;减法
273. DO\_SUB PROC
275. MOV AX,MTIME\_1
276. MOV BX,MTIME\_2
277. CMP AX,BX
278. JAE SUB\_NN
279. ;如果是负数先打印‘-’
280. XCHG AX,BX
281. SUB AX,BX
282. PRINTS SUB\_INFO
283. PRINTC '-'
284. PRINT\_NUMS AX
285. NEXTLINE
286. RET
287. ;是正数直接打印
288. SUB\_NN:
289. SUB AX,BX
290. PRINTS SUB\_INFO
291. PRINT\_NUMS AX
292. NEXTLINE
293. RET
295. DO\_SUB ENDP
297. ;乘法
298. DO\_MUL PROC
299. MOV AX,MTIME\_1
300. MOV BX,MTIME\_2
301. MUL BX
302. ;打印相乘结果
303. PRINTS MUL\_INFO
304. PRINT\_NUMS AX
305. NEXTLINE
306. RET
307. DO\_MUL ENDP
309. ;除法
310. DO\_DIV PROC
312. MOV AX,MTIME\_1
313. MOV BX,MTIME\_2
314. DIV BX
315. ;打印商
316. PRINTS DIV\_INFO
317. PRINT\_NUMS AX
318. ;打印余数
319. PRINTS MOD\_INFO
320. PRINT\_NUMS DX
321. NEXTLINE
322. RET
324. DO\_DIV ENDP
326. ;延时
327. DELAY PROC
329. ;读取时钟
330. MOV AH,00H
331. INT 1AH
332. MOV BX,DX
333. ;反复查询时间差额是否满足
334. REPTS:
335. MOV AH,00H
336. INT 1AH
337. MOV AX,DX
338. SUB AX,BX
339. ;0s<AX<1s
340. CMP AX,14
341. JL REPTS
342. CMP DELAY\_TIME,30
343. JBE DO\_REPE\_ADD
344. JA DO\_REPE\_SUB
345. ;延时时间反复游弋（8~32）
346. DO\_REPE\_ADD:
347. MOV BL,2
348. ADD DELAY\_TIME,BL
349. RET
350. DO\_REPE\_SUB:
351. MOV BL,24
352. SUB DELAY\_TIME,BL
353. RET
355. DELAY ENDP
357. CODES ENDS
358. END START

**运行截图：**

