

BCD

BCD	SBCD1	SBCD2	BCD	5
STACK	SEGMENT PARA	STACK	'STACK'	
	DB 200 DUP(?)			
STACK	ENDS			
DATA	SEGMENT			
SBCD1	DB 5, 6, 9, 2;2965			
SBCD2	DB 2, 3, 7, 8;8732			
SSUM	DB 5 DUP(0);	2965+8732=11697		
CONT	DB 4			
DATA	ENDS			
CODE	SEGMENT			
	ASSUME CS:CODE, DS:DATA, SS:STACK, ES:DATA			
SBCDAD:	MOV AX, DATA			
	MOV DS, AX			
	MOV ES, AX			
	CLC			
	CLD;DF=0 LODSB LODSB SI 1			
	MOV SI, OFFSET SBCD1			
	MOV DI, OFFSET SBCD2			
	MOV BX, OFFSET SSUM			
	MOV CL, CONT			
	MOV CH, 0			
	MOV AX, 0;AX			
SBCDAD1:	LODSB; [SI] AL SI++ DF=0			
	ADC AL, [DI]			
	AAA; BCD			
	INC DI			
	MOV BYTE PTR[BX], AL			
	INC BX			
	LOOP SBCDAD1			
	MOV BYTE PTR[BX], AL; SSUM			
	MOV AH, 4CH			
	INT 21H			

```
CODE    ENDS
        END SBCDAD
```

DEBUG

- : _u 0 29

```
-u0 29
0778:0000 B87707      MOV     AX,0777
0778:0003 8ED8       MOV     DS,AX
0778:0005 8EC0       MOV     ES,AX
0778:0007 F8        CLC
0778:0008 FC        CLD
0778:0009 BE0000     MOV     SI,0000
0778:000C BF0400     MOV     DI,0004
0778:000F BB0800     MOV     BX,0008
0778:0012 BA0E0D00   MOV     CL,[000D]
0778:0016 B500       MOV     CH,00
0778:0018 B80000     MOV     AX,0000
0778:001B AC        LODSB
0778:001C 1205       ADC     AL,[DI]
0778:001E 37        AAA
0778:001F 47        INC     DI
0778:0020 8807       MOV     [BX],AL
0778:0022 43        INC     BX
0778:0023 E2F6       LOOP    001B
0778:0025 8807       MOV     [BX],AL
0778:0027 B44C       MOV     AH,4C
0778:0029 CD21       INT     21
_
```

Figure 1:

0777, SSUM 0008

- : _g=0 27 MOV AH,4C
- : _d 0777:0

```

-----
-g=0 27

AX=0201 BX=000C CX=0000 DX=0000 SP=00C8 BP=0000 SI=0004 DI=0008
DS=0777 ES=0777 SS=076A CS=0778 IP=0027 NU UP EI PL NZ NA PE CY
0778:0027 B44C      MOV     AH,4C
-

```

Figure 2:

```

-d 0777:0
0777:0000 05 06 09 02 02 03 07 08-07 09 06 01 01 04 00 00 .....
0777:0010 B8 77 07 8E D8 8E C0 F8-FC BE 00 00 BF 04 00 BB .w.....
0777:0020 08 00 8A 0E 0D 00 B5 00-B8 00 00 AC 12 05 37 47 .....7G
0777:0030 88 07 43 E2 F6 88 07 B4-4C CD 21 C0 50 8D 86 7D ..C.....L!.P..}
0777:0040 FF 50 E8 89 69 83 C4 04-0B C0 75 8C B8 FF FF 50 .P..i.....u....P
0777:0050 B8 05 00 50 8D 86 7C FF-50 E8 94 12 83 C4 06 8B ...P..!P.....
0777:0060 5E 04 D1 E3 D1 E3 A1 3A-21 8B 16 3C 21 89 87 BE ^.....:!.<?...
0777:0070 22 89 97 C0 22 5E 8B E5-5D C3 55 8B EC 81 EC 86 "..."^..l.U.....

```

```

0777:0000 SSUM      0008, 9 13      11697

```

```

                ADD1 ADD2      ,      SUM      ,

STACK  SEGMENT PARA STACK 'STACK'
        DB 100 DUP(?)

STACK  ENDS
DATA   SEGMENT
ADD1   DB 0FEH, 86H, 7CH, 23H, 0A7H, 26H
ADD2   DB 0B3H, 73H, 11H, 9FH, 46H, 3EH
SUM     DB 6 DUP(0); :26A7237C86FEH+3E469F1173B3H=64EDC28DFAB1H
CONT    DB 3
DATA   ENDS
CODE   SEGMENT
        ASSUME CS:CODE, SS:STACK, DS:DATA, ES:DATA
MADDB: MOV AX, DATA
        MOV DS, AX
        MOV ES, AX
        MOV SI, OFFSET ADD1
        MOV DI, OFFSET ADD2
        MOV BX, OFFSET SUM
        MOV CL, BYTE PTR CONT
        MOV CH, 0
        CLC
MADDB1: MOV AX, [SI]
        ADC AX, [DI]
        INC SI
        INC SI
        INC DI

```

```

        INC DI
        MOV [BX], AX
        INC BX
        INC BX
        LOOP MADDB1
        MOV AH, 4CH
        INT 21H
CODE    ENDS
        END MADDB

```

DEBUG

- ```

 |_u 27
0773:0000 B87107 MOV AX,0771
0773:0003 8ED8 MOV DS,AX
0773:0005 8EC0 MOV ES,AX
0773:0007 BE0000 MOV SI,0000
0773:000A BF0600 MOV DI,0006
0773:000D BB0C00 MOV BX,000C
0773:0010 8A0E1200 MOV CL,[0012]
0773:0014 B500 MOV CH,00
0773:0016 F8 CLC
0773:0017 8B04 MOV AX,[SI]
0773:0019 1305 ADC AX,[DI]
0773:001B 46 INC SI
0773:001C 46 INC SI
0773:001D 47 INC DI
0773:001E 47 INC DI
0773:001F 8907 MOV [BX],AX
0773:0021 43 INC BX
0773:0022 43 INC BX
0773:0023 E2F2 LOOP 0017
0773:0025 B44C MOV AH,4C
0773:0027 CD21 INT 21

```
- ```

      :_u 0 29 |_▲
                0771, SUM      000C

```

```

      |_g=0 25
AX=64ED BX=0012 CX=0000 DX=0000 SP=0064 BP=0000 SI=0006 DI=000C
DS=0771 ES=0771 SS=076A CS=0773 IP=0025  NU UP EI PL NZ NA PE NC
0773:0025 B44C        MOV     AH,4C

```
- ```

 :_d 0771:0 |_
0771:0000 FE 86 7C 23 A7 26 B3 73-11 9F 46 3E B1 FA 8D C2 ..!#. &.s..F>...
0771:0010 ED 64 03 00 00 00 00 00-00 00 00 00 00 00 00 .d.....
0771:0020 B8 71 07 8E D8 8E C0 BE-00 00 BF 06 00 BB 0C 00 .q.....
0771:0030 8A 0E 12 00 B5 00 F8 8B-04 13 05 46 46 47 47 89 FFGG.
0771:0040 07 43 43 E2 F2 B4 4C CD-21 86 FA FE 2A E4 40 50 .CC...L.?.*..@P
0771:0050 8D 86 FA FE 50 8D 86 7C-FF 50 E8 C5 72 83 C4 06 P..!.P..r...
0771:0060 8B 9E 7A FF D1 E3 D1 E3-8B 87 CC 17 8B 97 CE 17 ..z.....
0771:0070 89 46 FC 89 56 FE 05 0C-00 52 50 E8 42 48 83 C4 .F..V....RP.BH..

```
- ```

      :_d 0771:0 |_
0771:000C 0771:0011 , 64EDC28DFAB1H

```

BUFFER 5 8

```

STACK    SEGMENT PARA      STACK    'STACK'
        DB 200 DUP(?)

STACK    ENDS
DATA     SEGMENT
BUFFER   DB 0A5H, 43H, 11H, 0C6H, 71H;    C6H A5H 71H 43H 11H
CONT     DB 4
DATA     ENDS
CODE     SEGMENT
        ASSUME CS:CODE, DS:DATA, SS:STACK, ES:DATA
SORT:    MOV AX, DATA;
        MOV DS, AX
        MOV ES, AX
        LEA DI, BUFFER
        MOV BL, CONT
        MOV AX, 0
        MOV DX, 0
NEXT0:   MOV SI, DI
        MOV CL, BL
NEXT2:   MOV AL, [SI]
        INC SI
        CMP AL, [SI]
        JNC NEXT3; AL>[SI]          [SI] [SI-1]
        MOV DX, [SI]
        MOV [SI-1], DL
        MOV [SI], AL
NEXT3:   DEC CL
        JNZ NEXT2;
        DEC BL
        JNZ NEXT0;
        MOV AH, 4CH
        INT 21H
CODE     ENDS
        END SORT

```

DEBUG

- ```

-u 0 31
0778:0000 B87707 MOV AX,0777
0778:0003 8ED8 MOV DS,AX
0778:0005 8EC0 MOV ES,AX
0778:0007 8D3E0000 LEA DI,[0000]
0778:000B 8A1E0500 MOV BL,[0005]
0778:000F B80000 MOV AX,0000
0778:0012 BA0000 MOV DX,0000
0778:0015 8BF7 MOV SI,DI
0778:0017 8ACB MOV CL,BL
0778:0019 8A04 MOV AL,[SI]
0778:001B 46 INC SI
0778:001C 3A04 CMP AL,[SI]
0778:001E 7307 JNB 0027
0778:0020 8B14 MOV DX,[SI]
0778:0022 8B54FF MOV [SI-01],DL
0778:0025 8B04 MOV [SI],AL
0778:0027 FEC9 DEC CL
0778:0029 75EE JNZ 0019
0778:002B FECB DEC BL
0778:002D 75E6 JNZ 0015
0778:002F B44C MOV AH,4C
0778:0031 CD21 INT 21

```
- ```

: _u 0 31
0777, BUFFER      0000

```
- ```

-g=0 2F
AX=00C6 BX=0000 CX=0100 DX=71C6 SP=00C8 BP=0000 SI=0001 DI=0000
DS=0777 ES=0777 SS=076A CS=0778 IP=002F NU UP EI PL ZR NA PE NC
0778:002F B44C MOV AH,4C

```
- ```

: _g=0 2F MOV AH, 4C

```
- ```

-d 0777:0
0777:0000 C6 A5 71 43 11 04 00 00-00 00 00 00 00 00 00 00 ..qC.....
0777:0010 B8 77 07 8E D8 8E C0 8D-3E 00 00 8A 1E 05 00 B8 .w.....>.....
0777:0020 00 00 BA 00 00 8B F7 8A-CB 8A 04 46 3A 04 73 07F:s.
0777:0030 8B 14 88 54 FF 88 04 FE-C9 75 EE FE CB 75 E6 B4 ...T...u...u...
0777:0040 4C CD 21 89 69 83 C4 04-0B C0 75 8C B8 FF FF 50 L.t.i...u...P
0777:0050 B8 05 00 50 8D 86 7C FF-50 E8 94 12 83 C4 06 8B ...P...i.P.....
0777:0060 5E 04 D1 E3 D1 E3 A1 3A-21 8B 16 3C 21 89 87 BE ^.....t...<?...
0777:0070 22 89 97 C0 22 5E 8B E5-5D C3 55 8B EC 81 EC 86 "...^...l.U....

```
- ```

: _d 0777:0
0777:0000 0777:0004 , C6H A5H 71H 43H 11H

```

ASCII

```

        ASCII      BINNUM      ASCBCD

STACK   SEGMENT PARA STACK 'STACK'
        DB 100 DUP(?)

STACK   ENDS

DATA    SEGMENT

BINNUM  DW 4FFFH;20479D

ASCD CD DB 5 DUP(0)

DATA    ENDS

CODE    SEGMENT

        ASSUME CS:CODE, SS:STACK, DS:DATA, ES:DATA

```

```

B2A:  MOV AX, DATA
      MOV DS, AX
      MOV ES, AX
      MOV CX, 5;4FFFH=20479D    5
      XOR DX, DX; DX
      MOV BX, 10
      MOV AX, BINNUM
      MOV DI, OFFSET ASCDCD
B2A1: DIV BX;AX/BX  AX, DX  AX      AX/10  AX
      ADD DL, 30H;DL+48 ASCII    '0' = 48D = 30H
      MOV [DI], DL
      INC DI
      AND AX, AX;CF
      CMP CX, 5
      JC  STOP; AX 0
      MOV DL, 0
      LOOP B2A1
STOP:  MOV AH, 4CH
      INT 21H
CODE  ENDS
      END B2A

```

DEBUG

```

      -u 0 27
0772:0000 B87107      MOV     AX,0771
0772:0003 8ED8      MOV     DS,AX
0772:0005 8EC0      MOV     ES,AX
0772:0007 B90500      MOV     CX,0005
0772:000A 33D2      XOR     DX,DX
0772:000C BB0A00      MOV     BX,000A
0772:000F A10000      MOV     AX,[0000]
0772:0012 BF0200      MOV     DI,0002
0772:0015 F7F3      DIV     BX
0772:0017 80C230      ADD     DL,30
0772:001A 8815      MOV     [DI],DL
0772:001C 47      INC     DI
0772:001D 23C0      AND     AX,AX
0772:001F 7404      JZ      0025
0772:0021 B200      MOV     DL,00
0772:0023 E2DB      LOOP    0000
0772:0025 B44C      MOV     AH,4C
0772:0027 CD21      INT     21
•   :_u 0 27 -
      0771, ASCBCD      0002

      -g=0 25
AX=0000 BX=000A CX=0001 DX=0032 SP=0064 BP=0000 SI=0000 DI=0007
DS=0771 ES=0771 SS=076A CS=0772 IP=0025  NU UP EI PL ZR NA PE NC
•   :_g=0 25 MOV AH, 4C  0772:0025 B44C      MOV     AH,4C

```

```

      -d 0771:0
0771:0000 FF 4F 39 37 34 30 32 00-00 00 00 00 00 00 00 00 .097402.....
0771:0010 B8 71 07 8E D8 8E C0 B9-05 00 33 D2 BB 0A 00 A1 .q.....3.....
0771:0020 00 00 BF 02 00 F7 F3 80-C2 30 88 15 47 23 C0 74 .....0..G#.t
0771:0030 04 B2 00 E2 F0 B4 4C CD-21 86 7A FF A1 70 08 39 .....L.!.z..p.9
0771:0040 86 7A FF 72 03 E9 8D 00-8A 86 FA FE 2A E4 40 50 .z.r.....*.eP
0771:0050 8D 86 FA FE 50 8D 86 7C-FF 50 E8 C5 72 83 C4 06 ....P..i.P..r...
0771:0060 8B 9E 7A FF D1 E3 D1 E3-8B 87 CC 17 8B 97 CE 17 ..z.....
0771:0070 89 46 FC 89 56 FE 05 0C-00 52 50 E8 42 48 83 C4 .F..U....RP.BH..

```

• : _d 0771:0-
97402 4FFFH=20749

ASCII 16

4 ASCII	ASCII	ASCSTF	BIN
STACK	SEGMENT PARA STACK 'STACK'		
	DB 100 DUP(?)		
STACK	ENDS		
DATA	SEGMENT		
BINNUM	DW 4FFFH;20479D		
ASCD CD	DB 5 DUP(0)		
DATA	ENDS		
CODE	SEGMENT		
	ASSUME CS:CODE, SS:STACK, DS:DATA, ES:DATA		
B2A:	MOV AX, DATA		
	MOV DS, AX		
	MOV ES, AX		
	MOV CX, 5;4FFFH=20479D	5	
	XOR DX, DX; DX		
	MOV BX, 10		
	MOV AX, BINNUM		
	MOV DI, OFFSET ASCD CD		
B2A1:	DIV BX;AX/BX	AX, DX	AX AX/10 AX
	ADD DL, 30H;DL+48 ASCII		'0' = 48D = 30H
	MOV [DI], DL		
	INC DI		
	AND AX, AX;CF		
	CMP CX, 5		
	JC STOP; AX 0		
	MOV DL, 0		
	LOOP B2A1		
STOP:	MOV AH, 4CH		
	INT 21H		
CODE	ENDS		
	END B2A		

DEBUG

- ```

 -u 0 27
0772:0000 B87107 MOV AX,0771
0772:0003 8ED8 MOV DS,AX
0772:0005 8EC0 MOV ES,AX
0772:0007 B90500 MOV CX,0005
0772:000A 33D2 XOR DX,DX
0772:000C BB0A00 MOV BX,000A
0772:000F A10000 MOV AX,[0000]
0772:0012 BF0200 MOV DI,0002
0772:0015 F7F3 DIV BX
0772:0017 80C230 ADD DL,30
0772:001A 8B15 MOV [DI],DL
0772:001C 47 INC DI
0772:001D 23C0 AND AX,AX
0772:001F 7404 JZ 0025
0772:0021 B200 MOV DL,00
0772:0023 E2DB LOOP 0000
0772:0025 B44C MOV AH,4C
0772:0027 CD21 INT 21

```
- ```

      :_u 0 34 -
      0771, ASCBCD      0002
      
```
- ```

 -g 0 25
 AX=0000 BX=000A CX=0001 DX=0032 SP=0064 BP=0000 SI=0000 DI=0007
 DS=0771 ES=0771 SS=076A CS=0772 IP=0025 NV UP EI PL ZR NA PE NC
 :_g=0 32 MOV AH, 4C 0772:0025 B44C MOV AH,4C

```
- ```

      -d 0771:0
0771:0000 FF 4F 39 37 34 30 32 00-00 00 00 00 00 00 00 00  .097402.....
0771:0010 B8 71 07 8E D8 8E C0 B9-05 00 33 D2 BB 0A 00 A1  .q.....3....
0771:0020 00 00 BF 02 00 F7 F3 80-C2 30 88 15 47 23 C0 74  .....0..6#.t
0771:0030 04 B2 00 E2 F0 B4 4C CD-21 86 7A FF A1 70 08 39  ....L.!.z..p.9
0771:0040 86 7A FF 72 03 E9 8D 00-8A 86 FA FE 2A E4 40 50  .z.r.....*.0P
0771:0050 8D 86 FA FE 50 8D 86 7C-FF 50 E8 C5 72 83 C4 06  ...P..!.P..r...
0771:0060 8B 9E 7A FF D1 E3 D1 E3-8B 87 CC 17 8B 97 CE 17  ..z.....
0771:0070 89 46 FC 89 56 FE 05 0C-00 52 50 E8 42 48 83 C4  .F..U....RP.BH..
      
```
- ```

 :_d 0771:0 -
 97402 4FFFH=20749

```