NAME: REIMARC G. CORPUZ DATE: 04/27/2023

COURSE & YEAR: BSCP 3 IE

|  |  |
| --- | --- |
| **CODE** | **SCREEN SHOTS** |
| .model small  .stack 100  .data  array db 0,1,2,3,4,5,6,7,8,9  max db 0  .code  mov ax, @data  mov ds, ax  xor di, di  mov cl, 10  lea bx, array  mov al, max  back:  cmp al, [bx+di]  jnc skip  mov dl, [bx+di]  mov al, dl  skip:  inc di  dec cl  jnz back  mov max, al  mov ch, 02h  mov cl, 04h  mov bh, al  mov ah, 4ch  int 21h  end | VARIABLE: |
| .model small  .stack 100h  .data  arr db 1,2,3,4,5    .code  mov ax, @data  mov ds, ax  mov cx, 5  mov si, 0  mov ah, 2  output:  mov dl, arr[si]  add dl, 30h  int 21h  inc si  loop output | OUTPUT: |
| .model small  .stack  .data  arr1 db 1h,2h,3h,4h,5h,6h,7h,8h,9h,10h,11h  arr2 db ''  .code  start:  mov ax, @data  mov ds, ax  mov al, 0  mov cx, 11  mov si, 0  labko:  add al, arr1[si]  mov arr2[si], al  inc si  loop labko  end start | VARIABLE: |
| .model small  .stack 100  .data  .code  MOV AX,1234H ;1  CALL ADDP  MOV BX,5678H ;5  MOV AH,4CH  INT 21H  ADDP PROC NEAR  INC AX ;2  CALL SUBP  NEG AX ;4  RET  ADDP ENDP  SUBP PROC NEAR  DEC AX ;3  RET  SUBP ENDP | CALL ADDP   * ADDP PROC NEAR     CALL SUBP   * SUBP PROC NEAR |
| .model small  .stack 100  .data  .code  MOV BX, 02  MOV AX, 01  MOV CX, 03H  CALL POWER  MOV AH, 4CH  INT 21H  POWER PROC NEAR  HI:  MUL BX  LOOP HI:  RET  POWER ENDP |  |
| .model small  .stack 100  .data  array db 5,1,9,3,8,6,7,4,2,0  num db 2  .code  mov ax, @data  mov ds, ax  xor di, di  mov cl, 10  lea bx, array  mov al, num  mov si, 0  back:  cmp al, [bx+di]  jnz skip  mov si, di  mov ah, 2  add si, 30h  mov dx, si  int 21h  skip:  inc di  dec cl  jnz back  mov ah, 4ch  int 21h  end |  |
| .model small  .data  hello db 'HELLO', 10,13,'4'  hello1 db ' ' ,10,13,'$'    .code  mov ax, @data  mov ds, ax  main:  mov si, 04h  mov cx, 05h  clc    add\_loop:  mov al, [hello+si]  or al, 20h  mov [hello+si], al  dec si  loop add\_loop    mov ah, 09  mov dx, offset hello1  int 21h | BEFORE:  AFTER: |
| .model small  .data  hello db 'abcde' ,10,13,'$'  hello1 db ' ' ,10,13,'$'  .code  mov ax, @data  mov ds, ax  main:  mov si, 04h  mov cx, 05h  clc    add\_loop:  mov al, [hello+si]  inc al  mov [hello1+si], al  dec si  loop add\_loop    mov ah, 09  mov dx, offset hello1  int 21h | BEFORE:    AFTER: |