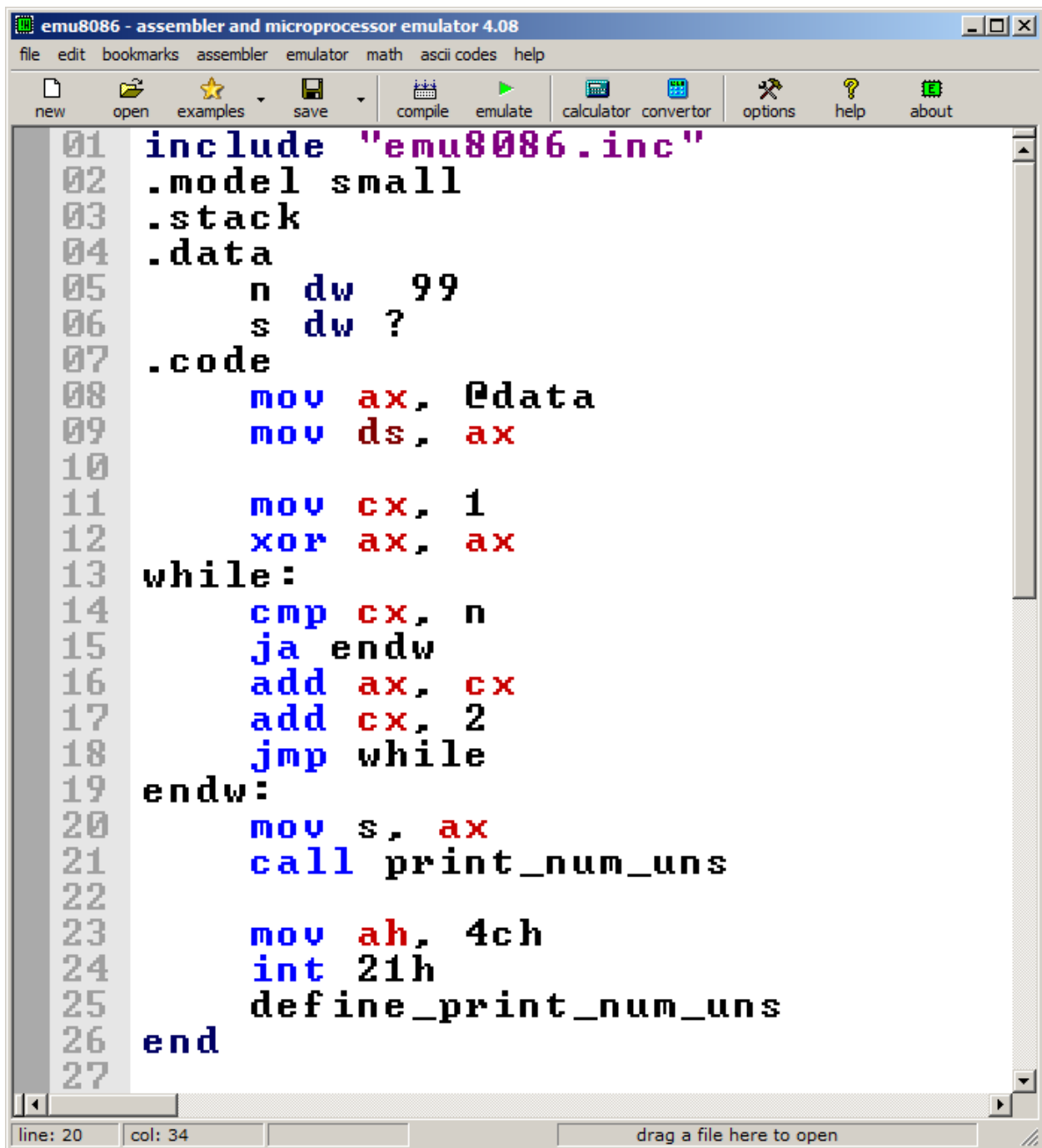
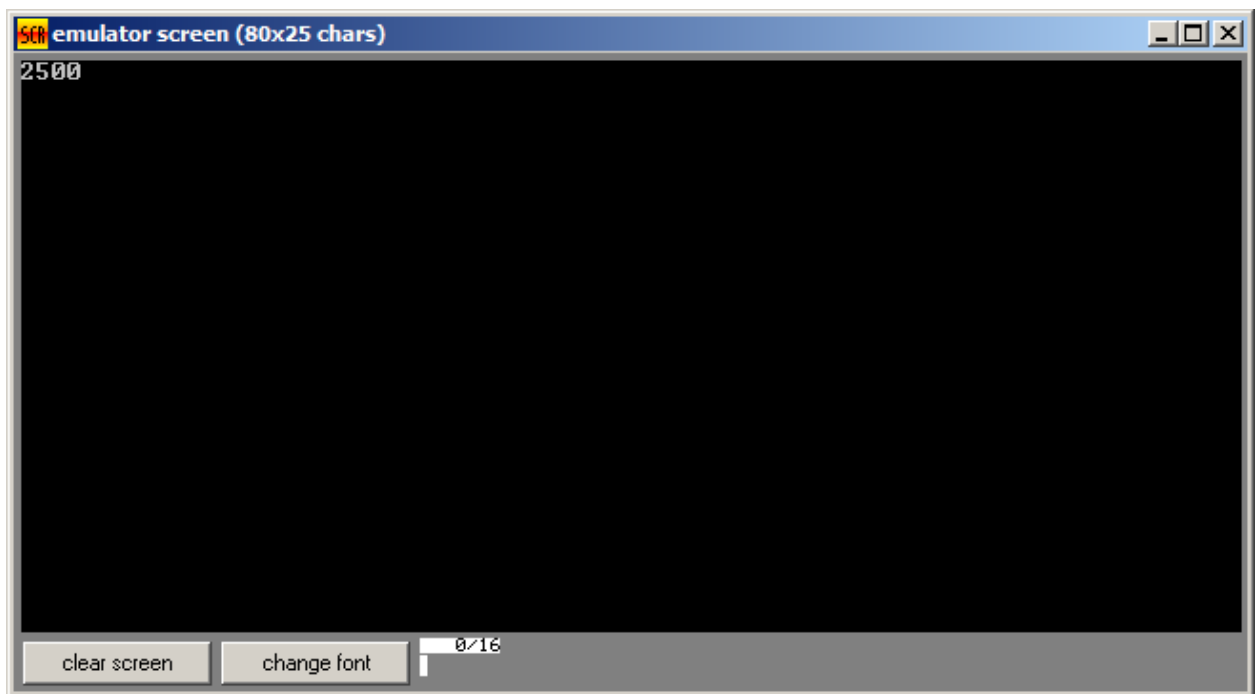


Tổng các số lẻ ≤ 99



```
01 include "emu8086.inc"
02 .model small
03 .stack
04 .data
05     n dw 99
06     s dw ?
07 .code
08     mov ax, @data
09     mov ds, ax
10
11     mov cx, 1
12     xor ax, ax
13 while:
14     cmp cx, n
15     ja endw
16     add ax, cx
17     add cx, 2
18     jmp while
19 endw:
20     mov s, ax
21     call print_num_uns
22
23     mov ah, 4ch
24     int 21h
25     define _print_num_uns
26 end
27
```

line: 20 col: 34 drag a file here to open



Giả lập phép nhân bằng phép cộng:

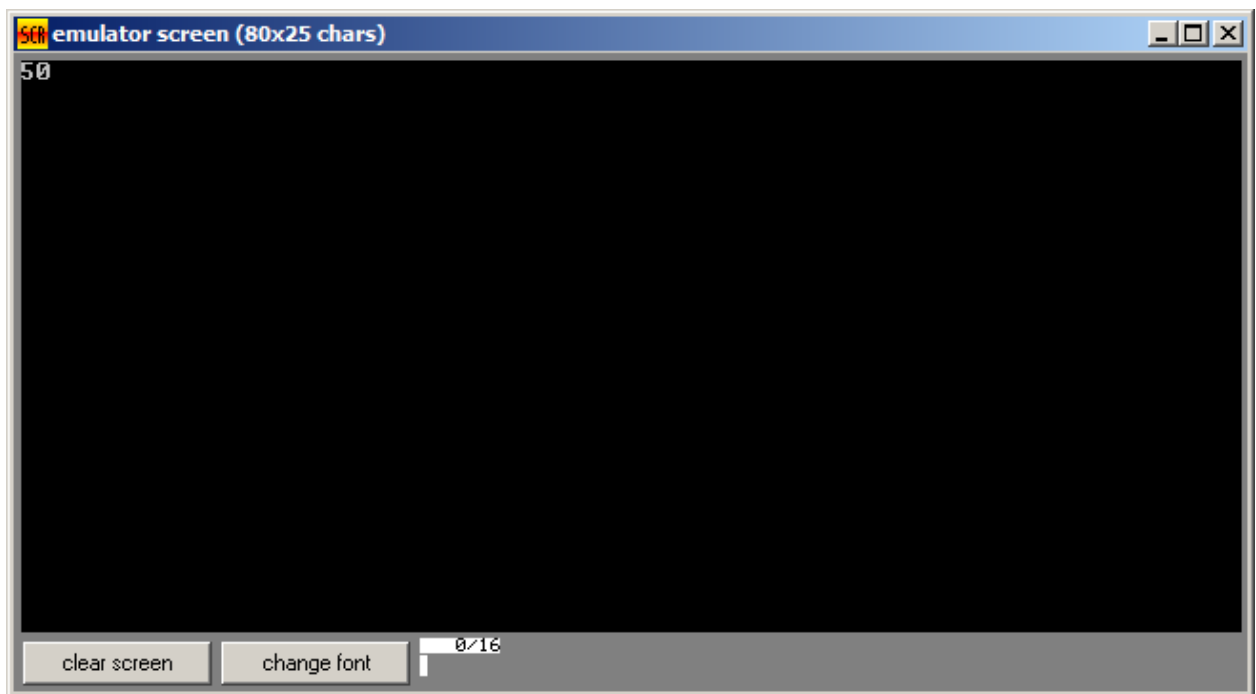
edit: D:_NOI\mycod110.asm

file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help about

```
01 include "emu8086.inc"
02 .model small
03 .stack
04 .data
05     a dw 10
06     b dw 5
07     s dw ?
08 .code
09     mov ax, @data
10     mov ds, ax
11
12     xor ax, ax
13     xor cx, cx
14 while:
15     cmp cx, b
16     jae endwhile
17     add ax, a
18     inc cx
19     jmp while
20 endwhile:
21     mov s, ax
22     call print_num_uns
23
24     mov ah, 4ch
25     int 21h
26     define_print_num_uns
27 end
```

line: 16 col: 15 drag a file here to open



Giả lập phép chia bằng phép trừ:

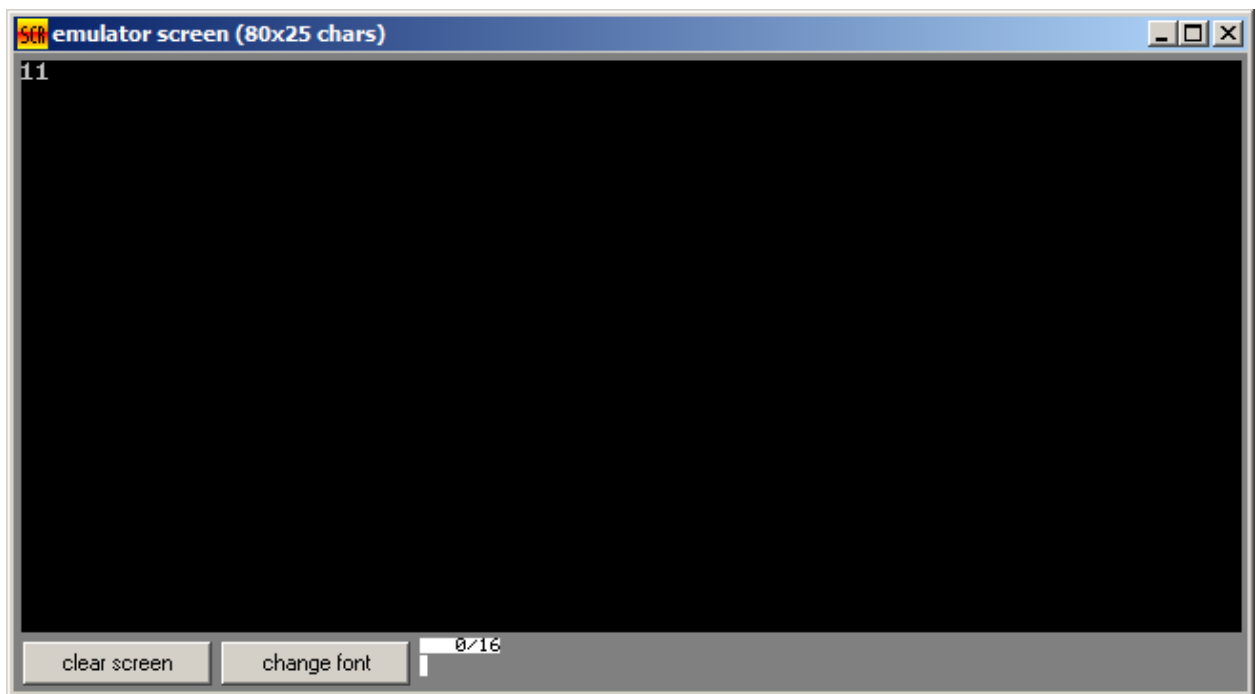
edit: D:_NOI\mycod111.asm

file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help about

```
01 include "emu8086.inc"
02 .model small
03 .stack
04 .data
05     a dw 56
06     b dw 5
07     c dw ?
08 .code
09     mov ax, @data
10     mov ds, ax
11
12     xor cx, cx
13     mov ax, a
14 while:
15     cmp ax, b
16     jb endw
17     inc cx
18     sub ax, b    |
19     jmp while
20 endw:
21
22     mov c, cx
23     mov ax, c
24     call print_num_uns
25
26     mov ah, 4ch
27     int 21h
```

line: 18 col: 17 drag a file here to open

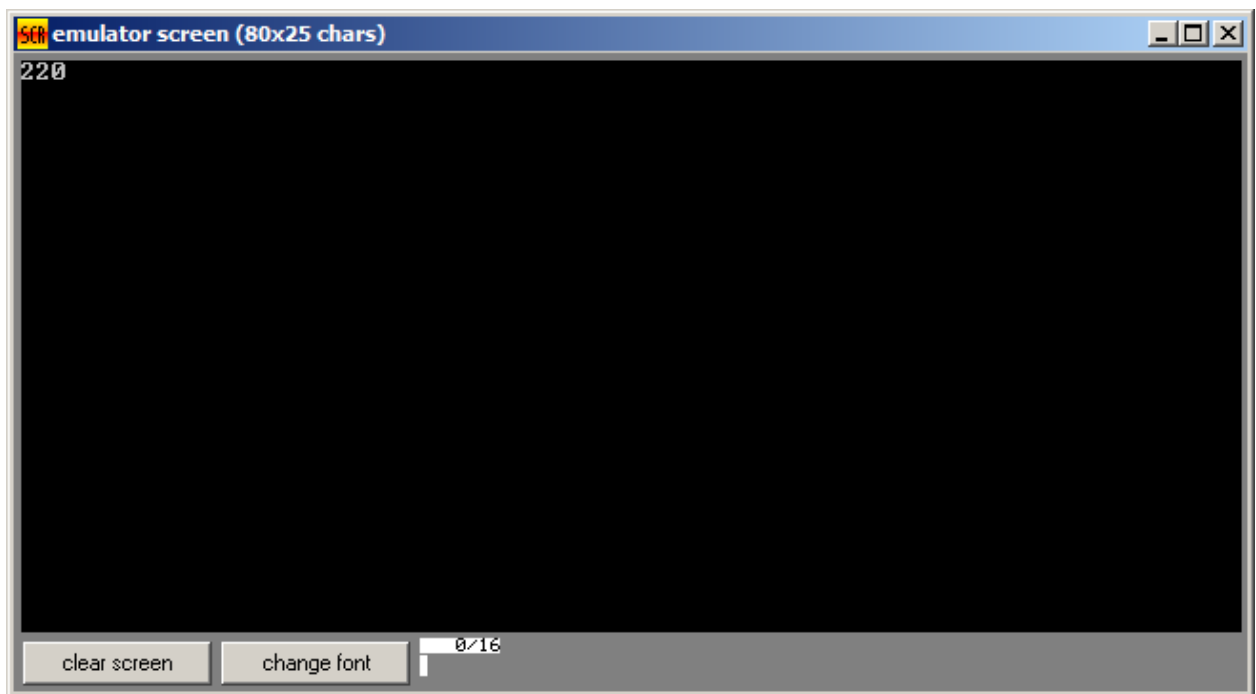


Bài 1d. $S(n)=1+(1+2)+(1+2+3)+\dots+(1+2+3+\dots+n)$.

```
edit: D:\_NOI\mycod112.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about

01 include "emu8086.inc"
02 .model small
03 .stack
04 .data
05     n dw 10
06     s dw 0
07     ;S(n)=1+(1+2)+(1+2+3)+...+(1+2+3+...+n)
08 .code
09     mov ax, @data
10     mov ds, ax
11     xor cx, cx
12     xor ax, ax
13     xor dx, dx
14 while:
15     cmp cx, n
16     ja endw
17     add ax, cx
18     inc cx
19     add dx, ax
20     jmp while
21 endw:
22     mov s, dx
23     mov ax, s
24     call print_num_uns
25     mov ah, 4ch
26     int 21h
27     define _print_num_uns
28 end
29
30
```

line: 10 col: 19 drag a file here to open



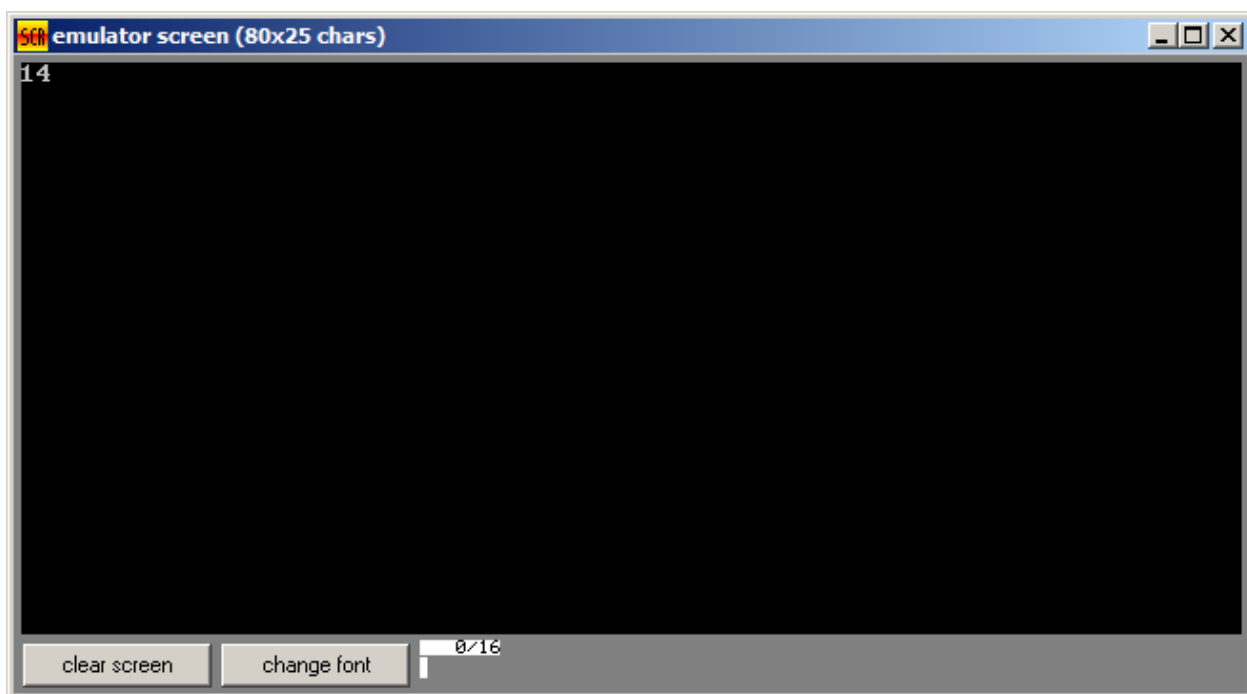
Bài 2a

$$S(n)=1^2+2^2+3^2+\dots+n^2 .$$

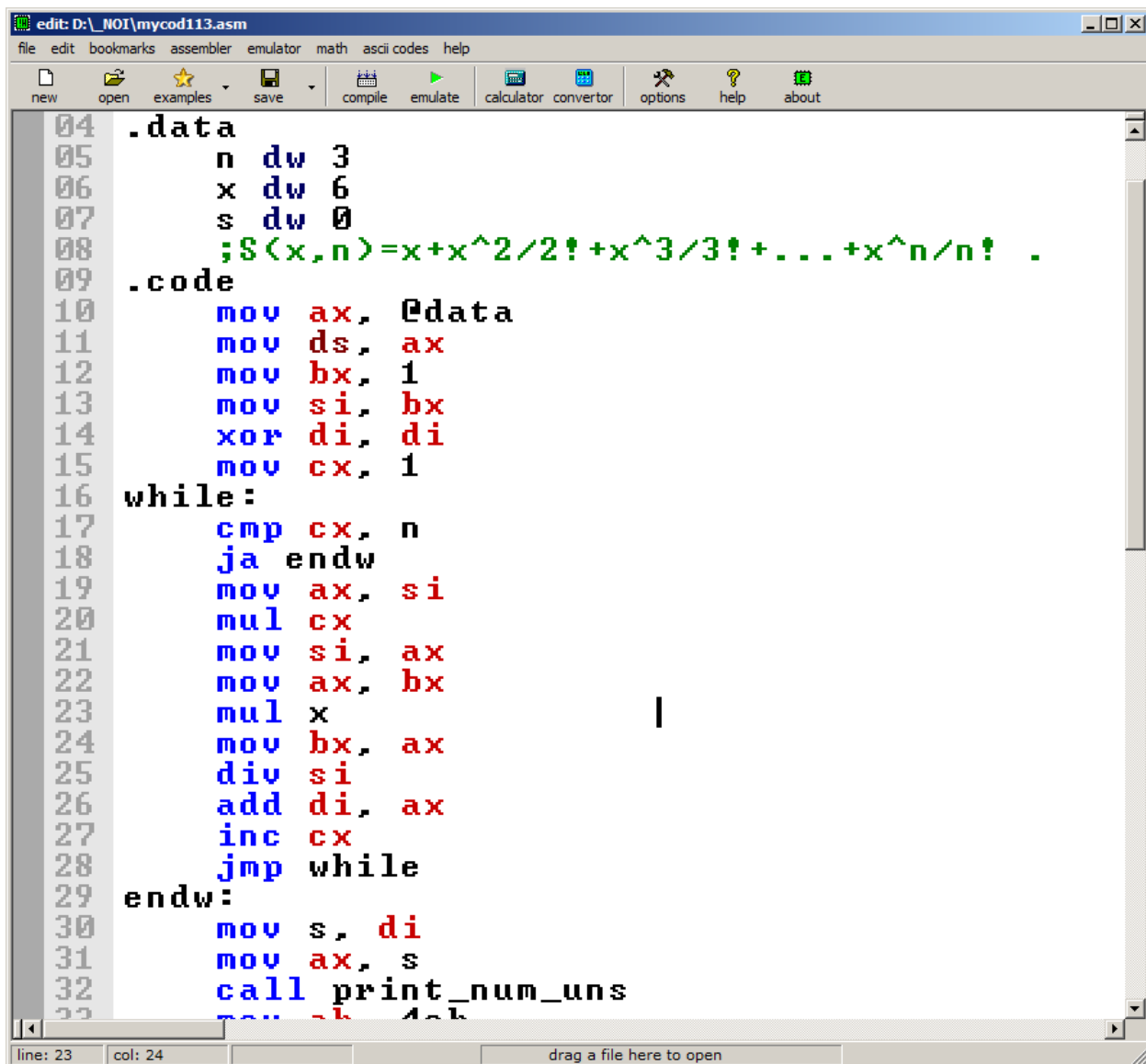

```
edit: D:\_NOI\mycod113.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about

01 include "emu8086.inc"
02 .model small
03 .stack
04 .data
05     n dw 3
06     s dw 0
07     ; S(n)=1^2+2^2+3^2+...+n^2
08 .code
09     mov ax, @data
10     mov ds, ax
11     mov cx, 1
12     xor ax, ax ; binh phuong
13     xor si, si ; si
14 while:
15     cmp cx, n
16     ja endw
17     mov ax, cx
18     mul cx ; dx thay doi noi dung
19     inc cx
20     add si, ax
21     jmp while
22 endw:
23     mov s, si
24     mov ax, s
25     call print_num_uns
26     mov ah, 4ch
27     int 21h
28     define _print_num_uns
29 end
30
```

line: 21 col: 31 drag a file here to open



Bài 5f. $S(x,n)=x+x^2/2!+x^3/3!+\dots+x^n/n!$.



```
edit: D:\_NOT\mycod113.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about

04 .data
05     n dw 3
06     x dw 6
07     s dw 0
08     ;S(x,n)=x+x^2/2!+x^3/3!+...+x^n/n!
09 .code
10     mov ax, @data
11     mov ds, ax
12     mov bx, 1
13     mov si, bx
14     xor di, di
15     mov cx, 1
16 while:
17     cmp cx, n
18     ja  endw
19     mov ax, si
20     mul cx
21     mov si, ax
22     mov ax, bx
23     mul x
24     mov bx, ax
25     div si
26     add di, ax
27     inc cx
28     jmp while
29 endw:
30     mov s, di
31     mov ax, s
32     call print_num_uns
33     mov ax, 40h
```

line: 23 col: 24 drag a file here to open

