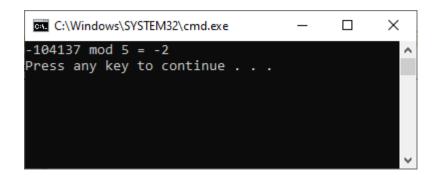
<u>Apeluri de sistem 7</u>. Se dau două numere naturale a și b (a: DWORD, b: WORD, definite în segmentul de date). Să se calculeze a/b și să se afișeze restul împărțirii în următorul format: "<a> mod = <rest>". Exemplu: pentru

a = 23 si b = 5 se va afișa: "23 mod 5 = 3" Valorile vor fi afișate în format decimal (baza 10) cu semn.

```
Code
Data
                             mov
                                             ; AX = b = 5
fstr DB "%d mod %d = %d", 0
                                   AX, [b]
a DD -104137
                             cwde
                                             ; EAX = b = 5
b DW 5
                                  EBX, EAX
                                            ; EBX = b = 5
                             mov
                             mov
                                   EAX, [a]
                                             ; EAX = a = -104137
                                              ; EDX:EAX = a = -104137
                             cdq
                             push EAX
                                             ; backup a
                                             ; EAX = a/b = -20827, EDX = a\%b = -2
                             idiv EBX
                             pop EAX
                                             ; restore a
                             ; printf("%d mod %d = %d",a,b,{EDX} = a%b)
                             push EDX
                                               ; push arg result = a%b
                                               ; push arg b
                             push EBX
                                               ; push arg a
                             push EAX
                             push fstr
                                               ; push format str
                             call [printf]
                                                ; call printf
                             add ESP, 4*4
                                               ; clean stack (EDX + a + b + fstr) => 4 DWORDS
```



Operații fișiere 24. Se dau un nume de fișier și un text (definite în segmentul de date). Textul conține litere mici, litere mari, cifre și caractere speciale. Să se înlocuiască toate CIFRELE din textul dat cu caracterul 'C'. Să se creeze un fișier cu numele dat și să se scrie textul obținut prin înlocuire în fișier.

```
Data
                              Code
fmode DB "w", 0
                              mov ESI, text ; ESI = first(text)
fname DB "file.txt", 0
                              mov EDI, ESI ; EDI = ESI = first(text)
text DB "mD0rC_95_hqp$Rt;423", 0
                                            ; DF = 0 (frst->lst)
                              cld
                              parse:
                                  lodsb
                                           ; AL = text[i] = *(ESI), ESI++
                                  or AL, AL ; AL |= AL; cmp AL, 0; set ZF
                                  jz fin ; text[i]=='\0', end of text
                                  cmp AL, '0' ; AL - '0', set flags
                                  jb .write ; if (AL<'0'), don't change char</pre>
                                  cmp AL, '9'; AL - '9', set flags
                                  ja .write ; if (AL>'9'), don't change char
                                  mov AL, 'C' ; if AL in '0'...'9', AL='C'
                                  .write:
                                            ; text[i] = *(EDI) = AL, EDI++, now EDI=ESI
                                  stosb
Result:
                                  jmp parse ; while (text[i]!='\0')
                              fin:
file.txt
"mDCrC CC hqp$Rt;CCC"
                              ; fopen ("file.txt", 'w')
                              push fmode ; push offset fmode "w" - write
                              add ESP, 2*4; clean stack
                              ; fprintf(fptr, {modified_}text)
                              ; push fptr
                              push EAX
                              call [fprintf] ; call fprintf
                              add ESP, 2*4 ; clean stack
```