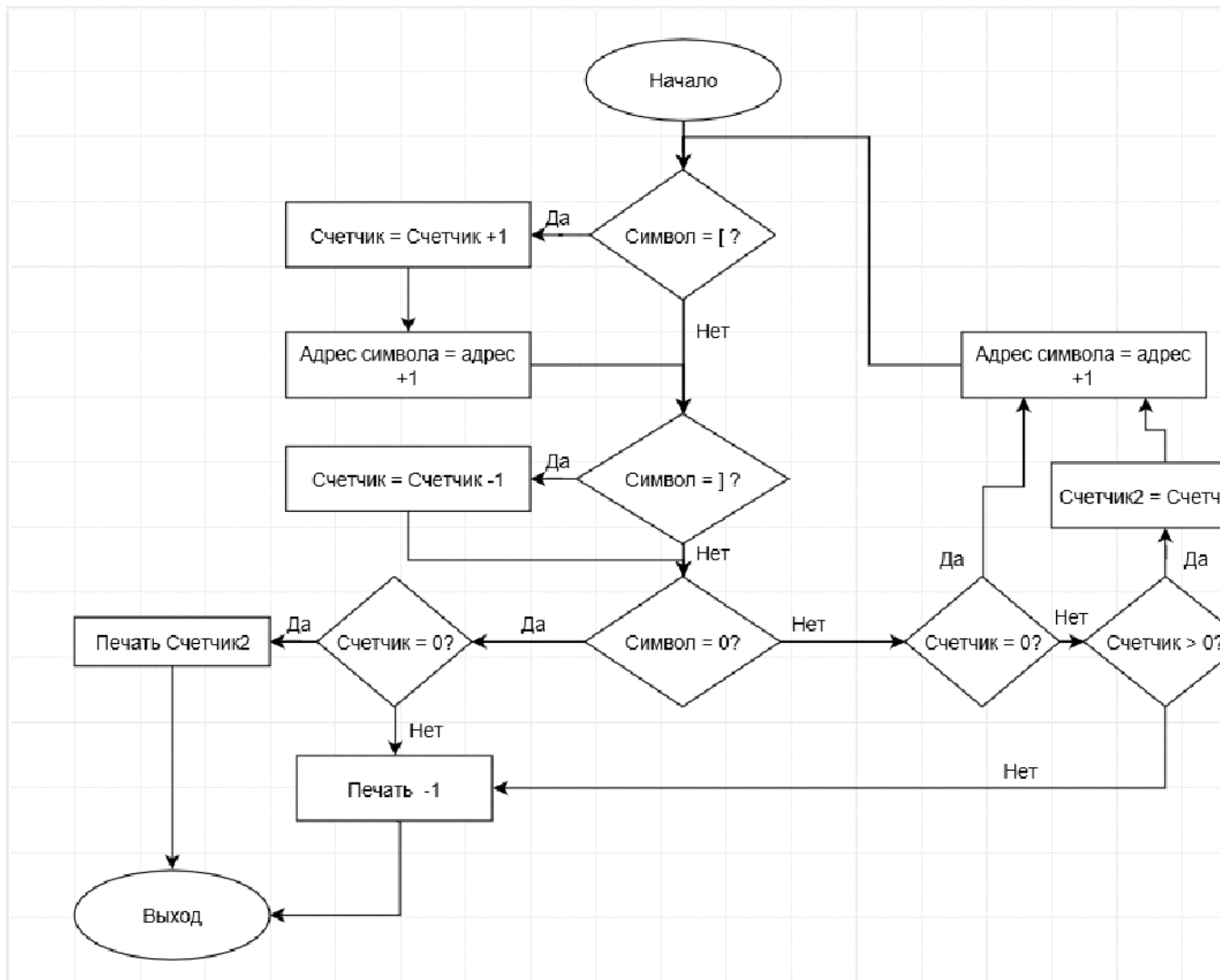


Задание 1 (12)

Найти количество символов, заключенных в квадратные скобки  
(вернуть -1, если скобки расставлены неправильно)

•



Алгоритм программы

- Код программы
- `%include "io.inc"`
- `section .text`
- `global CMAIN`
- `CMAIN:`
- `xor        eax, eax`
- `lea        ebx, [A]`
- `true:`

- mov       al, [ebx]
- cmp       eax, "["
- jne       if
- inc       dword [k]
- inc       ebx
- if:       cmp       eax, "]"
- jne       srv
- dec       dword [k]
- srv:       mov       ecx, dword [k]
- cmp       eax, 0
- jne       nxt
- cmp       ecx, 0
- je       quit
- fal:       PRINT\_DEC   2, -1
- jmp       Q
- nxt:       cmp       ecx, 0
- jl       fal
- je       eq
- inc       dword [n]
- eq:       inc       ebx
- jmp       true
  
- quit:
- mov       eax, dword [n]
- PRINT\_DEC 4, eax
- Q:
- ret
  
- section .data
- k: dd 0
- n: dd 0
- A: db "1+[32+2\*0]",0

- Трассировочная таблица

A: db "1+[3-2]"

A: db ""]12["

№	Элемент (ebx)	eax	k	n
5-6	1	0	0	0
8-28	1	«1»	-	-

8-28	2	«+»	=	=
8-11	3	«[»	1	-
12	4	«3»	-	-
12-2 8	4	-	-	1
8-28	5	«-»	-	2
8-28	6	«2»	-	3
8-20	7	«]»	0	-
29	-	3	-	-
Итог: Вывод «3»				
№	Элемент (ebx)	eax	k	n
5-6	1	0	0	0
8-2 4	1	«]»	-1	-
21	1	-	-	-
33	-	-	-	-
Итог: Вывод «-1»				

## Задание 2 (11)

Удалить из массива элементы таким образом, чтобы разность соседних элементов была не менее 5

- Алгоритм программы

Начало Алг

i=адрес конца массива

Вызов Прог

Прог:

Если i=адрес начала массива,

То Прыжок Конец Алг

Конец Если

Если значение(i) > значение(i-1)

То Разность=Значение(i)-Значение(i-1)

Иначе Разность=Значение(i-1)-Значение(i)

```

        Конец Если
        Если Разность > 5,
            То Сохраним i
                Цикл по i от i до адрес конца массива
                    Значение(i-1)=Значение(i)
                    i=i+1
                Конец цикл
            Загрузим i
        Конец Если
        i=i-1
        Прыжок Прог
Конец Алг

```

- Код программы

- section .text
- global CMAIN
- CMAIN:
  - lea ebx, [A]
  - mov eax, dword [N]
  - shl eax, 2
  - add ebx, eax
  - call Alg
  - ret
- Alg:
  - lea eax, [A]
  - cmp eax, ebx
  - je quit
  - mov eax, [ebx]
  - mov edx, [ebx-4]
  - cmp eax, edx
  - jl ls
  - sub eax, edx
  - jmp nxt
  - ls: sub edx, eax
  - mov eax, edx

- `nxt:           cmp eax, 5`
- `j1 ngr`
- `lea ecx, [A]`
- `mov eax, dword [N]`
- `shl eax,2`
- `add ecx, eax`
- `push ebx`
- `agn:   mov eax, [ebx]`
- `mov [ebx-4], eax`
- `add ebx, 4`
- `cmp ebx, ecx`
- `j1 agn`
- `pop ebx`
- `ngr:   sub ebx, 4`
- `dec dword [N]`
- `jmp Alg`
- `quit:`
- `ret`
  
- `section .data`
- `N: dd 3`
- `A: dd 6,9,3,8`

- Трассировочная таблица

#	ebx	eax	A	N
5-9	4202512	12	6,9,3,8	3
11-37	4202508	8	6,9,8	2
11-37	4202504	9	6,9,8	2
11-40	4202500	6	6,9,8	2
12	-	-	-	-
Итог: Остался массив 6,9,8 где все соседние элементы не больше 5				