

```
;
; Ucitava se broj N, zatim se ucitavaju clanovi niza i smestaju u
niz u rastucem poretku. Clanove koji vec postoje u nizu treba
zanemariti.
;
```

```
%include "asm_io.inc"
segment .data
```

```
poruka1 db "Unesite broj N: ",0
poruka2 db "Unesite clan niza ",0
```

```
segment .bss
```

```
niz resd 100
N resd 1
x resd 1
```

```
segment .text
```

```
global _asm_main
_asm_main:
```

```
enter 0,0 ; setup routine
pusha
```

```
mov eax, poruka1
call print_string
call read_int
mov [N], eax ; ukupan broj clanova
mov ecx, 0 ; brojac trenutnih clanova niza
```

```
mov eax, poruka2
call print_string
call read_int
mov [niz], eax
inc ecx
mov esi, 4
```

```
formiranje_niza:
```

```
cmp ecx, [N]
je kraj ; popunili smo ceo niz
mov eax, poruka2
call print_string
call read_int
mov [x], eax ; potencijalni clan
```

```
mov esi, 0
mov ebx, 0
```

```
;proveravamo da li je clan vec deo niza
postojanje_clana:
```

```
    cmp ebx, ecx
    je uvrsti
    mov eax, [x]
    sub eax, [niz+esi]
    cmp eax, 0

    je formiranje_niza
    add esi, 4
    inc ebx
    jmp postojanje_clana
```

uvrsti:

```
    mov eax, ecx
    dec eax ; pocinjemo proveru od poslednjeg clana
    mov ebx, 4
    mul ebx
    mov esi, eax
```

pp:

```
    mov eax, [niz+esi]
    ;dump_regs 1
    sub eax, [x]
    shl eax, 1
    jc stavi
    mov ebx, [niz+esi] ; pravimo prostor novom clanu niza
    mov [niz+esi+4], ebx
    sub esi, 4
    cmp esi, 0
    jb stavi
    jmp pp
```

stavi:

```
    mov ebx, [x]
    mov [niz+esi+4], ebx
    inc ecx
    jmp formiranje_niza
```

kraj:

```
    mov esi, 0
    mov ebx, 0
```

stampanjeniza:

```
    cmp ebx, ecx
    je krajzadatka
    mov eax, [niz+esi]
    call print_int
    call print_nl
    inc ebx
    add esi, 4
    jmp stampanjeniza
```

krajzadatka:

```
popa
mov    eax, 0          ; return back to C
leave
ret
```