

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

2. Számjegyek

13 pont

A következő algoritmus egy N jegyű ($N \geq 2$) szám összes számjegyének egyszeri felhasználásával készíthető számok közül meghatározza az eredeti számnál nagyobb, legkisebb számot. Ha nem létezik ilyen szám, akkor az eredeti számot írja ki.

A szám jegyeit az alábbi karakter típusú vektorban tárolja és kezeli:

Változó: `szam[0..N]` Karakter. Kódolja az algoritmust a választott programozási nyelven!

Az elkészült program forráskódját mentse NLSz néven!

A megoldás során vegye figyelembe a következőket:

- A "szam" változó (paraméter) típusát a feladatkiírás tartalmazza.
- A választott programozási nyelvtől függően eltérő jelölésű operátorokat, adattípusokat és függvényeket kell alkalmaznia.
- A "Térj vissza" utasítás megszakítja a függvény futását és meghatározza annak visszatérési értékét.

Konstans $N=5$

Függvény `Cserel(szam, i1:Egész, i2:Egész):Tömb[0..N] Karakter`

Változó `ch:Karakter`

`ch:=szam[i1]`

`szam[i1]:=szam[i2]`

`szam[i2]:=ch`

`Cserel:=szam`

Függvény vége

Függvény `Keres(szam): Tömb[0..N] Karakter`

Változó `indA, indB, i, j, meddig:Egész`

`indA:=-1`

`i:=N-1`

Ciklus amíg $i > 0$ és $indA = -1$

Ha `szam[i-1] < szam[i]` akkor

`indA:=i-1`

Elágazás vége

`i:=i-1`

Ciklus vége

Ha $indA = -1$ akkor

Térj vissza `szam`

Elágazás vége

`indB:=indA+1`

Ciklus $j:=indA+2$ -től $N-1$ -ig (+1 lépésközzel)

Ha `szam[j] > szam[indA]` és `szam[j] < szam[indB]` akkor

`indB:=j`

Elágazás vége

Ciklus vége

`szam:=Cserel(szam, indA, indB)`

Ciklus $meddig:=N-1$ -től 1 -ig (-1 lépésközzel)

Ciklus $j:=indA+1$ -től $meddig-1$ -ig (+1 lépésközzel)

Ha `(szam[j] > szam[j+1])` akkor

`Cserel(szam, j, j+1)`

Elágazás vége

Ciklus vége

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Ciklus vége
Keres:=szam
Függvény vége

Program NLSz:

```
Változó szam /*típusa a feladatkiírásban*/  
szam[0]='5'  
szam[1]='2'  
szam[2]='6'  
szam[3]='3'  
szam[4]='1'  
Ki: szam  
Ki: Keres(szam)  
Program vége.
```