



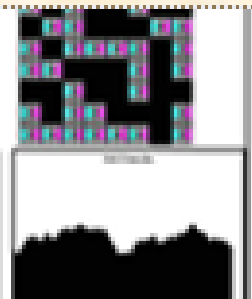
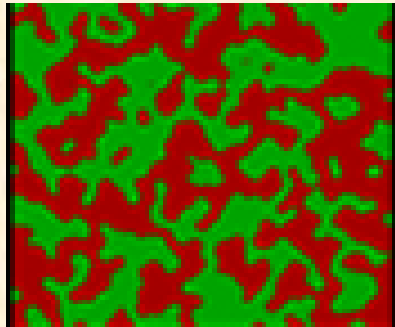
Nagypontosságú aritmetika I.

Nagypontosságú aritmetika



Problémák:

- sokjegyű (100 vagy 1000 vagy ...) egész számok kellenek több alkalmazásban;
- jó lenne, ha $1/3^*3$ értéke 1 lenne, azaz kellenének racionális számok, esetleg nagyon sok számjegyű számlálóval és nevezővel;
- ha beszélünk fixpontos számábrázolásról, jó lenne ha lennének fixpontos valós számok;
- nagy pontosságú lebegőpontos valós számok kellenek több alkalmazásban.



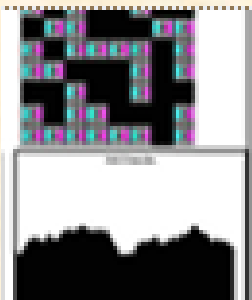
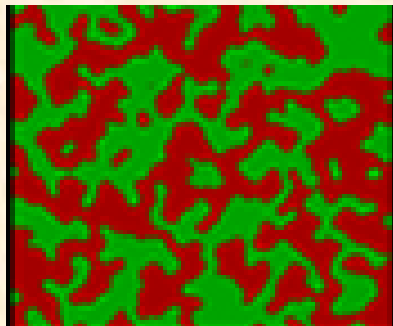
Nagypontosságú aritmetika: egész számok



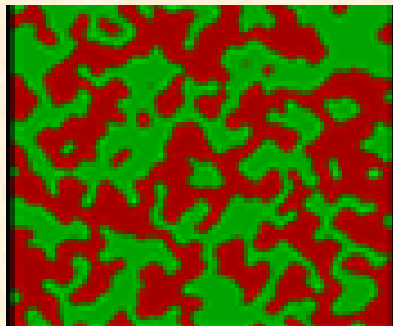
Számok ábrázolása:

- komplementes ábrázolás (negatív számok így nagyon sokjegyűek)
- előjel + számjegyek + hossz + számrendszer (tömb vagy szöveg):

$$x = \pm t_0 + t_1S + t_2S^2 + \dots + t_nS^n$$



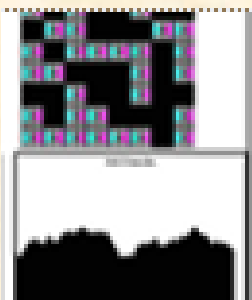
Nagypontosságú aritmetika: egész számok



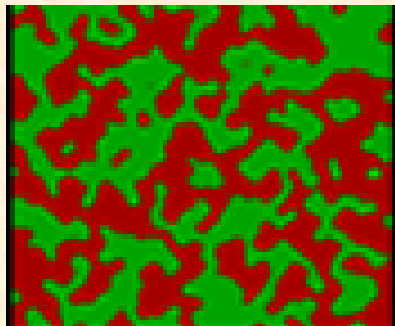
NagyEgész típus:

- előjel: $\{-, +\}$
- n: Egész
- S: alapszám
- T: tömb(0..Maxn:Egész)

$$x = \pm t_0 + t_1S + t_2S^2 + \dots + t_nS^n$$



Nagypontosságú aritmetika: egész számok



A műveleteknél az előjelet külön kezeljük, a műveleteket visszavezetjük pozitív számokkal végzett műveletekre.

Összeadás (x, y, z) :

`z.előjel:=x.előjel; z.S:=x.S`

Ha `x.előjel=y.előjel`

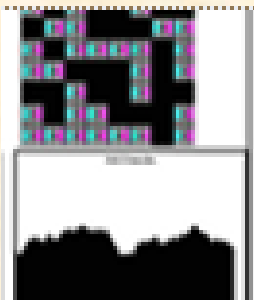
akkor `Összead(x, y, z)`

különben ha `x.előjel<y.előjel`

akkor `Kivon(y, x, z)`

különben `Kivon(x, y, z)`

Eljárás vége.



Nagypontosságú aritmetika: egész számok



Még arra is figyelhetünk, hogy $|x| \geq |y|$ legyen!

Összeadás (x, y, z) :

$z.\text{előjel} := x.\text{előjel}; z.S := x.S$

Ha $x.\text{előjel} = y.\text{előjel}$

akkor Ha nagy (x, y) akkor $\text{Összead}(x, y, z)$

különben $\text{Összead}(y, x, z)$

különben ha $x.\text{előjel} = '-'$

akkor Ha nagy (y, x) akkor $\text{Kivon}(y, x, z)$

$z.\text{előjel} := '+'$

különben $\text{Kivon}(x, y, z)$

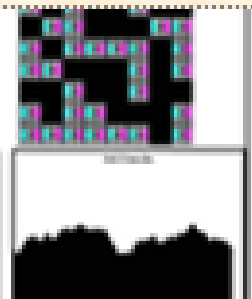
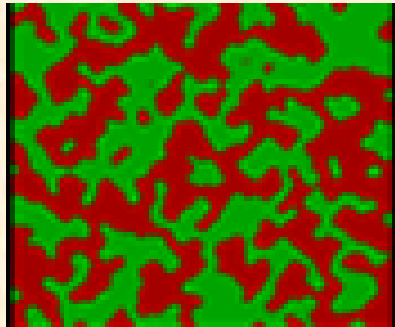
különben ha nagy (x, y) akkor $\text{Kivon}(x, y, z)$

különben $\text{Kivon}(y, x, z)$

$z.\text{előjel} := '-'$

Eljárás vége.

A belső eljárások pozitív számokat használnak!



Nagypontosságú aritmetika: egész számok



Összeadás, ahogyan az iskolában tanultuk!

$$\begin{array}{r} \text{XXXXXX} \\ + \quad \text{XXXX} \\ \hline \text{XXXXXX} \\ \uparrow \uparrow \uparrow \uparrow \uparrow \uparrow \end{array}$$

Összeadás (x, y, z) :

ÁT:=0

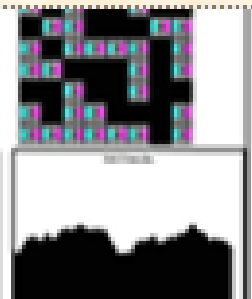
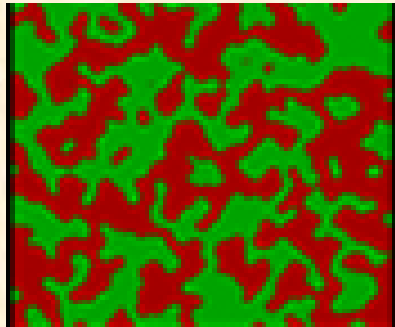
Ciklus $i=0$ -tól $y.N$ -ig

$a := (x.t(i) + y.t(i) + \text{ÁT})$

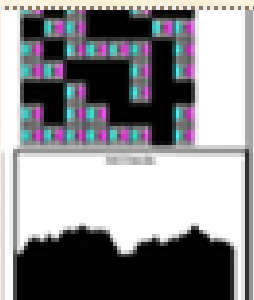
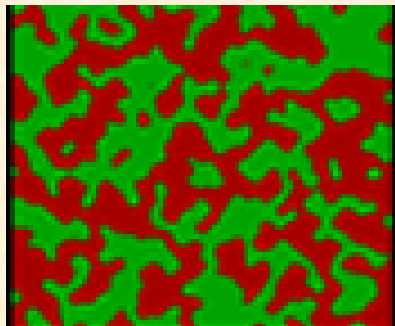
$z.t(i) := a \bmod x.S; \text{ÁT} := a \text{ div } x.S$

Ciklus vége

...



Nagypontosságú aritmetika: egész számok



...

$i := y.N + 1$

Ciklus amíg $\text{ÁT} \neq 0$

$a := (x.t(i) + \text{ÁT})$

$z.t(i) := a \bmod x.S; \text{ÁT} := a \operatorname{div} x.S; i := i + 1$

Ciklus vége

Ciklus amíg $i \leq x.N$

$z.t(i) := x.t(i); i := i + 1$

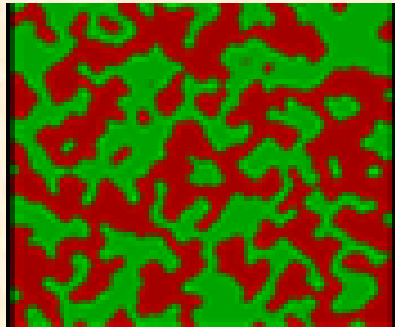
Ciklus vége

$z.N := i - 1$

Eljárás vége.

Probléma: a div és a \bmod művelet lassú!

Nagypontosságú aritmetika: egész számok



Felhasználhatjuk, hogy az átvitel maximum 1 lehet, így az osztást elkerülhetjük.

A megoldás:

$$a := (x.t(i) + y.t(i) + \text{ÁT})$$

$$z.t(i) := a \bmod x.S; \text{ÁT} := a \operatorname{div} x.S$$

helyett

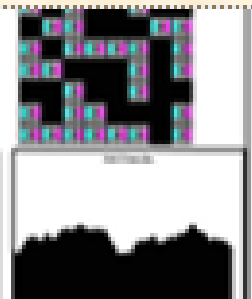
$$a := (x.t(i) + y.t(i) + \text{ÁT})$$

$$\text{Ha } a < x.S \text{ akkor } z.t(i) := a; \text{ÁT} := 0$$

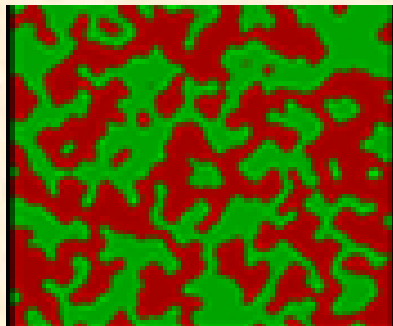
$$\text{különben } z.t(i) := a - x.S; \text{ÁT} := 1$$

Ugyanezt a második ciklusmaggal is megtehetjük.

A harmadik ciklus az algoritmus elején egy tömb-értékadással helyettesíthető: $z := x!$

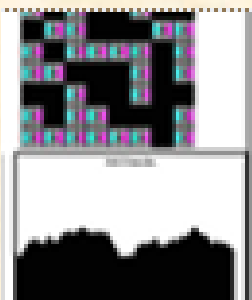


Nagypontosságú aritmetika: egész számok

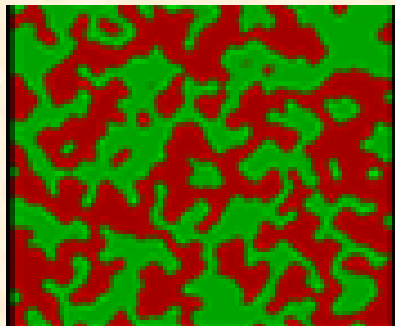


Megoldásváltozatok:

- kivonás – $z:=x-y$;
- eggyel növelés – $z:=x+1$;
- eggyel csökkentés – $z:=x-1$;
- növelés helyben – $x:=x+y$;
- csökkentés helyben – $x:=x-y$;
- eggyel növelés helyben – $x:=x+1$;
- eggyel csökkentés helyben – $x:=x-1$;
- ...



Nagypontosságú aritmetika: egész számok



Szorzás, ahogyan az iskolában tanultuk!

$$\begin{array}{r} \text{XXXX} * \text{XXX} \\ \hline \end{array}$$

$$\text{XXXXX}$$

$$\text{XXXXXX}$$

$$\text{XXXXXXXX}$$

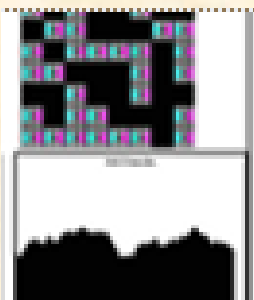
$$\text{XXXXXXXXXX}$$



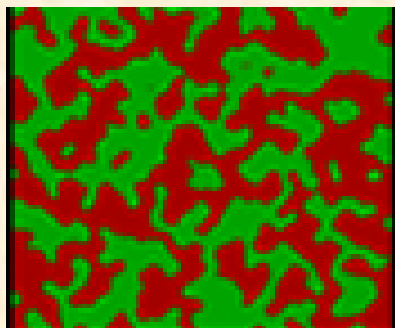
Probléma: a közbülső adatoknak mennyi helyet kell foglalnunk?

Megoldás: amint egy sort kiszámoltunk, azonnal hozzáadhatjuk az eredményhez.

Újabb probléma: az eredmény minden számjegyével sok műveletet végzünk.



Nagypontosságú aritmetika: egész számok



Megoldás egy különleges sorrendű kiszámolás!

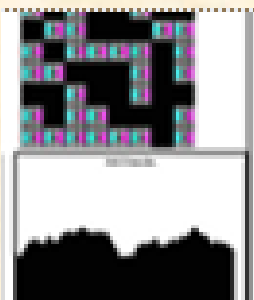
$$\underline{X \ X \ X \ X^* \ XXX}$$

$$XXX_4 X_2 X_1$$

$$XXXX_5 X_3$$

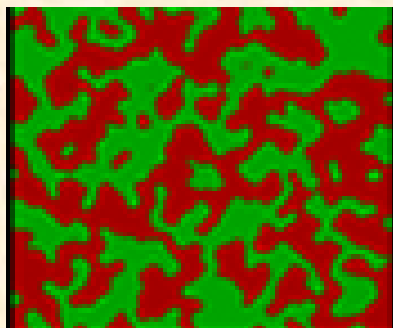
$$\frac{XXXXXX_6}{XXXXXXX \ X \ X}$$

Amikor egy oszlopot számolunk, azt azonnal adjuk hozzá a végeredményhez!



$$z.t(i) := \left(\sum_{j+k=i} x.t(j) * y.t(k) + \acute{A}T \right) \text{ mod } x.S$$

Nagypontosságú aritmetika: egész számok



A képlet átfogalmazva:

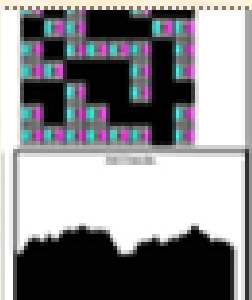
$$z.t(i) := \left(\sum_{j=?}^? x.t(j) * y.t(i - j) + \acute{A}T \right) \bmod x.S$$



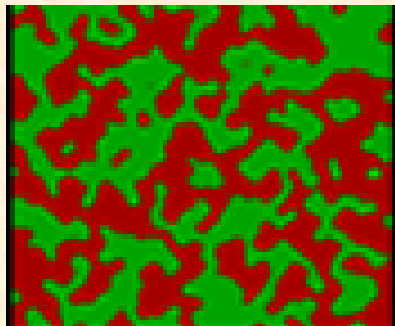
Az indexhatárok számítása:

$$0 \leq j \leq x.N \text{ és } 0 \leq i - j \leq y.N$$

$$\rightarrow z.t(i) := \left(\sum_{j=\max(0, i-y.N)}^{\min(i, x.N)} x.t(j) * y.t(i - j) + \acute{A}T \right) \bmod x.S$$



Nagypontosságú aritmetika: egész számok



Szoroz (x, y, z) :

Ciklus $i=0$ -tól $x.N+y.N+1$ -ig

$a := \text{ÁT}$

Ciklus $j=\max(0, i-y.N)$ -től $\min(i, x.N)$ -ig

$a := a + x.t(j) * y.t(i-j)$

Ciklus vége

$z.t(i) := a \bmod x.S$; $\text{ÁT} := a \text{ div } x.S$

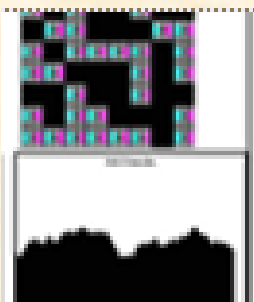
Ciklus vége

Ha $\text{ÁT}=0$ akkor $z.N := x.N+y.N$

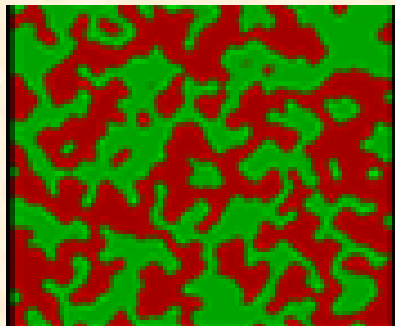
különben $z.N := x.N+y.N+1$; $z.t(z.N) := \text{ÁT}$

Eljárás vége.

Megjegyzés: az alapszám hatványaival szorzást
persze nem így oldjuk meg, hanem a kitevővel
eltolással.



Nagypontosságú aritmetika: egész számok



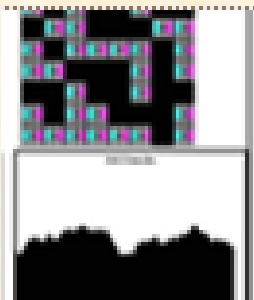
Felezéses szorzási algoritmus kettes számrend-
szerhez:

$$A * B = \begin{cases} A & \text{ha } B = 1 \\ (2 * A) * (B/2) & \text{ha } B \text{ páros} \\ A + A * (B - 1) & \text{ha } B \text{ páratlan} \end{cases}$$



Felezéses hatványozási algoritmus kettes szám-
rendszerhez:

$$A^B = \begin{cases} A & \text{ha } B = 1 \\ (A * A)^{(B/2)} & \text{ha } B \text{ páros} \\ A * A^{(B - 1)} & \text{ha } B \text{ páratlan} \end{cases}$$



Kérdések:

- páros-páratlan?
- $2 * A$, $B / 2$?
- $B - 1$?

Nagypontosságú aritmetika: egész számok



A megoldás:

Sorozat (A, B, C) :

$C := 0$

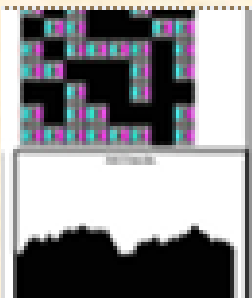
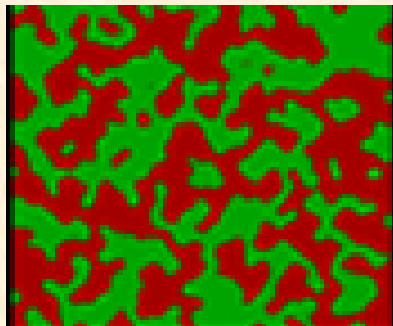
Ciklus amíg $B \neq 0$

Ha páros (B) akkor Balraléptet (A)
Jobbraléptet (B)
különben Össze (C, A)
Csökkent (B)

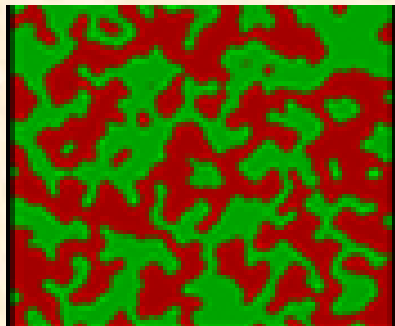
Ciklus vége

Eljárás vége.

Hasonló a hatványozási algoritmus is!



Nagypontosságú aritmetika: egész számok



Osztás ($C := A \text{ div } B$, $D := A \text{ mod } B$)

Naiv módszerrel:

Osztás (A, B, C, D) :

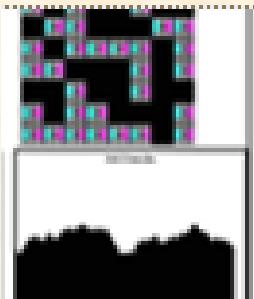
$C := 0$; $D := A$

Ciklus amíg $D \geq B$

$C := C + 1$; $D := D - B$

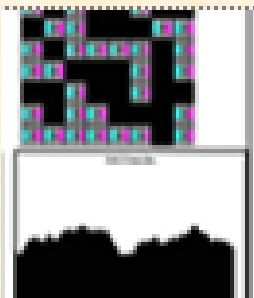
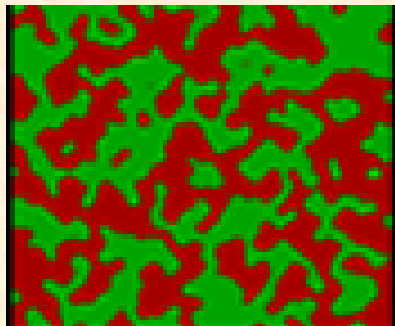
Ciklus vége

Eljárás vége.



Jól működne, de mennyi idő alatt?

Nagypontosságú aritmetika: egész számok



XXXXXXXXXX

- XXXX

- XXXX

... ←

XXXXXXXXXX

- XXXX

- XXXX

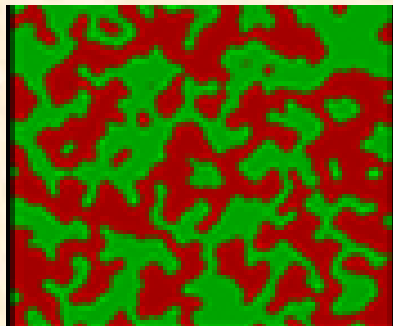
... →

0XXXXXXXXXX

- XXXX

...

Nagypontosságú aritmetika: egész számok



Osztás ($C := A \text{ div } B$, $D := A \text{ mod } B$)

Az osztandó balra tolása, kivonása, majd 1-1 lépéssel jobbra tolása a következő kivonások előtt:

Osztás (A, B, C, D):

$K := A.n - B.n$; $D := A$; $C := 0$

$v := \text{Balraléptet}(B, K)$;

Ciklus $i=1$ -től K -ig

 Balraléptet(C)

 Ciklus amíg $D \leq v$

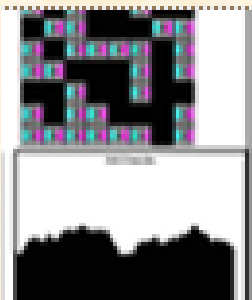
$C := C + 1$; $D := D - v$

 Ciklus vége

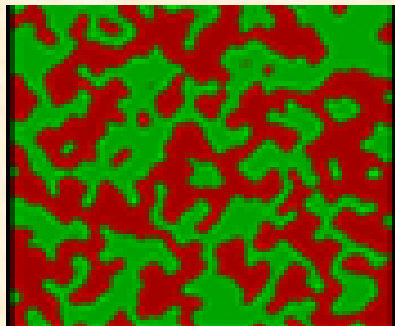
 Jobbraléptet(v)

Ciklus vége

Eljárás vége.



Nagypontosságú aritmetika: egész számok



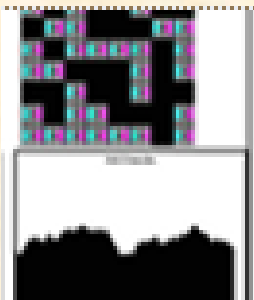
Hányados becslésével, visszavezetés "n+1-jegyű osztása n-jegyűvel" esetre: $(u_0, u_1, \dots, u_{n+1}) / (v_0, \dots, v_n)$.
A H hányados Q becslése a következő ($v \geq S/2$ esetén): $Q := \frac{S u_{n+1} + u_n}{v_n}$, így $Q = H$ vagy $H+1$ vagy $H+2$.



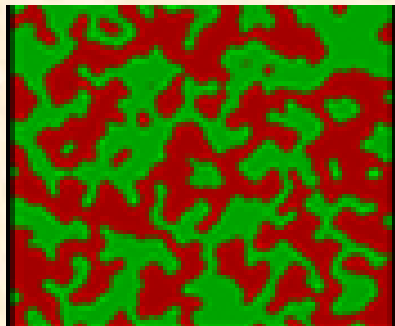
Egy gyors ellenőrzési lehetőség:

$$v_{n-1} * Q > (S * u_{n+1} + u_n - Q * v_n) * S + u_{n-1}$$

Ha igaz, akkor $Q := Q - 1$ (esetleg kétszer)



Nagypontosságú aritmetika: egész számok



Osztás1 (...):

Ha $U(N+1) = V(N)$ akkor $Q := S - 1$

különben $Q := (S * U(N+1) + U(N)) / V(N)$

Ciklus amíg $V(N-1) * Q >$

$(S * U(N+1) + U(N) - Q * V(N)) * S + U(N-1)$

$Q := Q - 1$

Ciklus vége

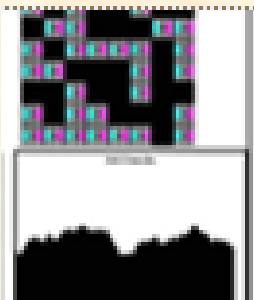
$W() := V() * Q$

$U() := U() - W()$

Ha $U() < 0$ akkor $U() := U() + V() : Q := Q - 1$

$H := Q$

Eljárás vége.



Nagypontosságú aritmetika: egész számok



Relációk: kisebb ($<$) reláció

Kisebb(A,B) :

Ha $A.előjel \neq B.előjel$

akkor $Kisebb := A.előjel = '-'$

különben

Ha $A.előjel = '-'$ akkor $csere(A,B)$

Ha $A.n \neq B.n$ akkor $Kisebb := A.n < B.n$

különben $i := A.n$

Ciklus amíg $i \geq 0$ és

$A.t[i] = B.t[i]$

$i := i - 1$

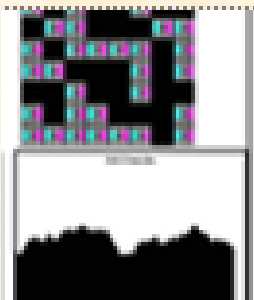
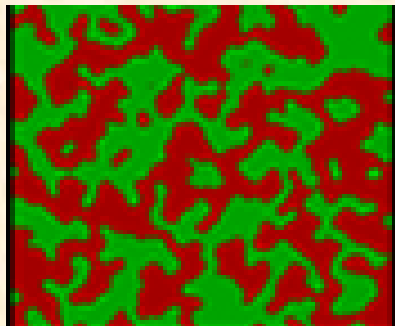
Ciklus vége

$Kisebb := i \geq 0$ és $A.t[i] < B.t[i]$

Elágazás vége

Elágazás vége

Eljárás vége.



A high-angle, wide shot of a modern building's courtyard. The building's facade is a vibrant red, composed of a grid of square panels. Many of these panels are replaced by windows of various sizes, some with white frames and others with dark frames. The courtyard floor is a light, neutral color, and the building's structure is visible from above, showing the grid pattern extending across the courtyard. In the center of the image, there is a semi-transparent yellow rectangular box containing the word "Vége" in a black, serif font.

Vége