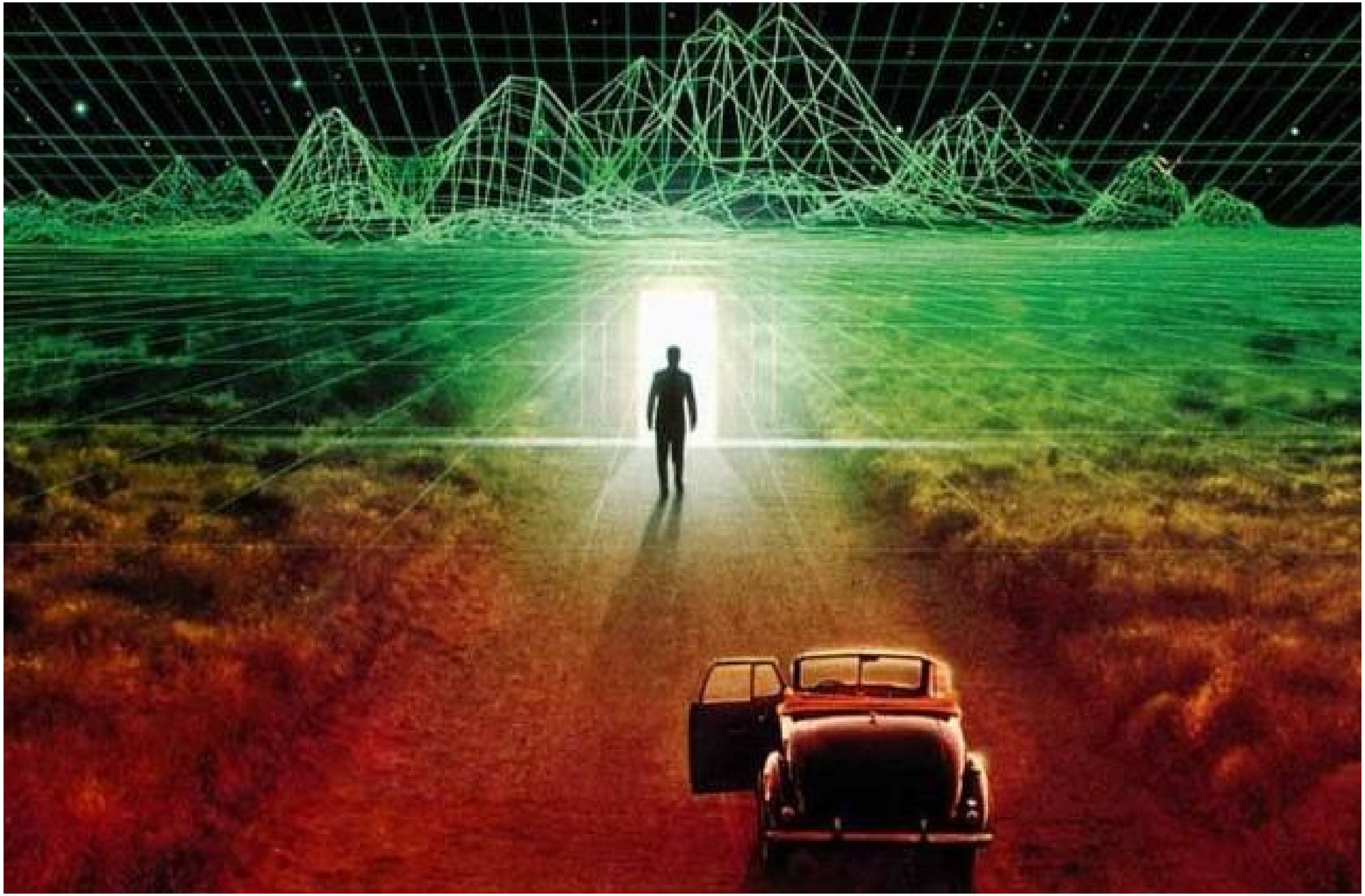
"КОМПЬЮТЕРНАЯ АЛГЕБРА"

Нормальная форма Смита

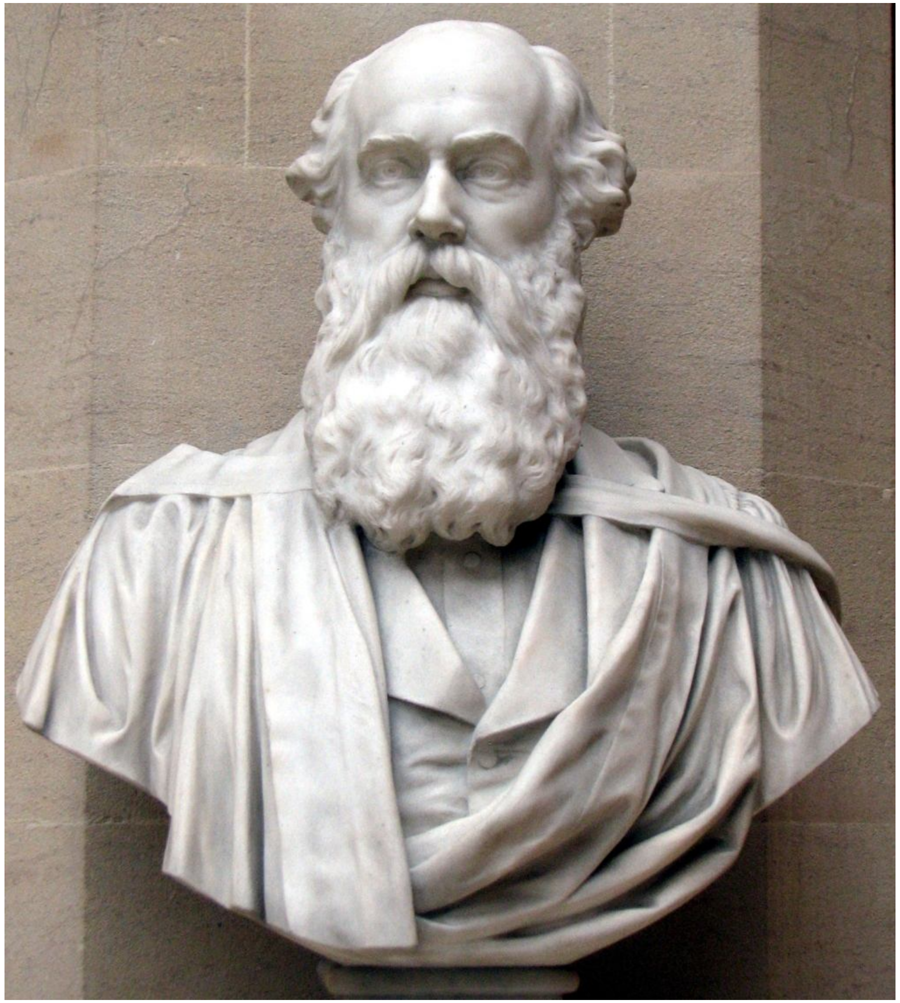


Henry John Stephen Smith

(**1826 – 1883**) английский математик, специалист в области теории чисел и теории матриц, изобретатель так называемой ***нормальной*** (или ***канонической***) ***формы Смита*** для матриц **над кольцом целых чисел**.

Henry John Stephen Smith,
British mathematician.

Statue on permanent display in the **Oxford University** Museum of Natural History.



The Cantor set

- *was* **discovered** *in* **1874** *by* **Henry John Stephen Smith**

and **introduced** *by* **Georg Cantor** *in* **1883**.

ЦЕЛОЧИСЛЕННЫЕ МАТРИЦЫ
и элементарные преобразования над ними

$A \in \mathrm{Mat}(m, n, \mathbb{Z})$; $A - -\!\!-\!\!\!\longrightarrow A'$

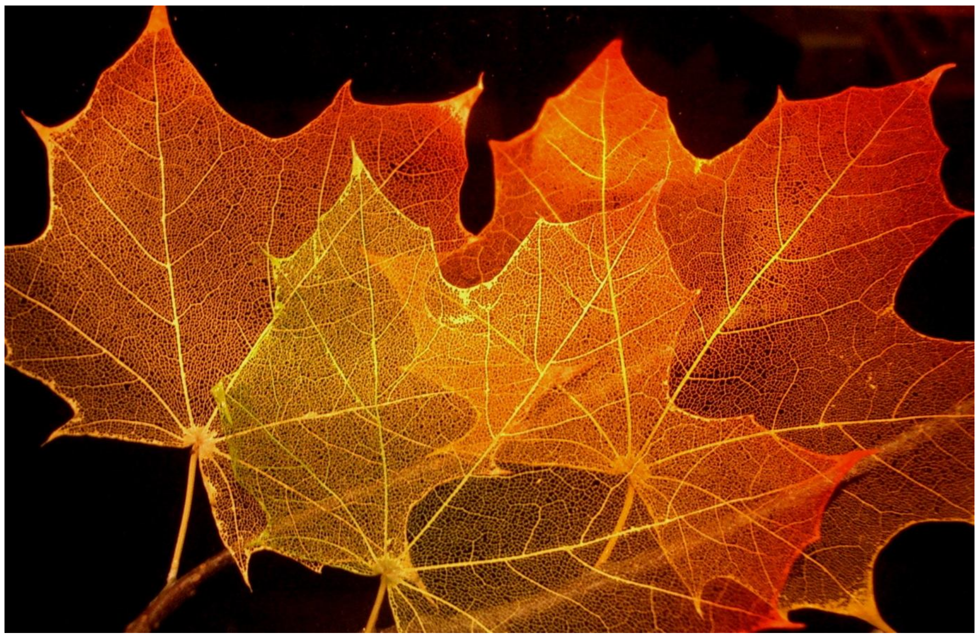
Тип	Над строками	Над столбцами	*Примечание*
I	$i^{\text{стр}} \leftrightarrow j^{\text{стр}}$	$i^{\text{стб}} \leftrightarrow j^{\text{стб}}$	$i \neq j$
II	$i^{\text{стр}} + j^{\text{стр}} \cdot c$	$i^{\text{стб}} + j^{\text{стб}} \cdot c$	$1 \leq i, j \leq m$; $i \neq j$; $c \in \mathbb{Z}$
III	$i^{\text{стр}} \cdot c$	$i^{\text{стб}} \cdot c$	$c \in \mathbb{Z}^*$

На **Maple-языке** (пакет **LinearAlgebra**)

```
RowOperation(A,[i,j])        ColumnOperation(A,[i,j])

RowOperation(A,[i,j],c)      ColumnOperation(A,[i,j],c)

RowOperation(A,i,c)          ColumnOperation(A,i,c)
```



Теорема 1 (Г. Смит, 1861). ***Произвольную целочисленную матрицу $\boldsymbol{A}$ с помощью элементарных преобразований над строками и столбцами можно привести к следующему (однозначно определенному) виду:***

$$\boldsymbol{S} = \left[\begin{array}{cccc|cccc} \boldsymbol{\mu_1} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & \boldsymbol{\mu_2} & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \boldsymbol{\mu_r} & 0 & 0 & \cdots & 0 \\ \hline 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{array}\right]$$

где $\boldsymbol{r} = \mathbf{rank}(A)$; $\boldsymbol{\mu_1}, \boldsymbol{\mu_2}, \dots, \boldsymbol{\mu_r} \in \mathbb{N}$; $\boldsymbol{\mu_i} \mid \boldsymbol{\mu_{i+1}} (\boldsymbol{i} = \mathbf{1}, \dots, \boldsymbol{r} - \mathbf{1})$.

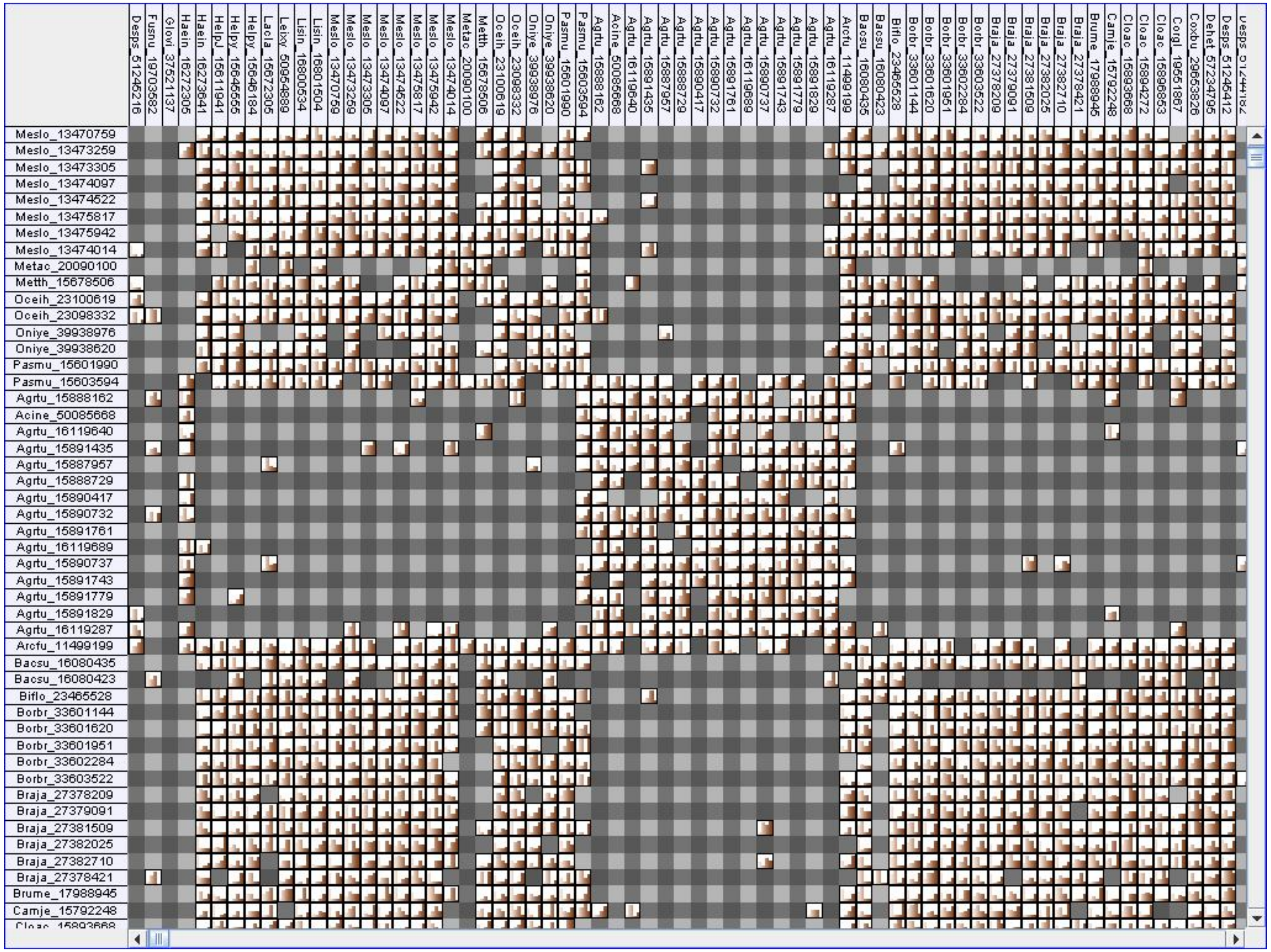
Дополнительная информация. Инвариантные множители $\boldsymbol{\mu_1, \mu_2, \ldots, \mu_r}$ могут быть выражены через **НОДМ**'ы, т. е. **наибольшие общие делители миноров:**

$$d_k = \text{НОД}\left\{A\begin{pmatrix} i_1 i_2 \cdots i_k \\ j_1 j_2 \cdots j_k \end{pmatrix} - \text{все миноры порядка } k\right\};$$

$k = 1, \ldots, r$; полагается также: $d_0 = 1$.

Формулы, выражающие μ_k через d_k:

$$\mu_k = \frac{d_k}{d_{k-1}}.$$


Desps_51245216
Fusnu_19703682
Glovi_37521137
Haein_16272305
Haein_16273641
Helpj_15611941
Helpy_15645555
Helpy_15646184
Laela_15672305
Leibx_50954889
Lisin_16800534
Lisin_16801504
Meslo_13470759
Meslo_13473259
Meslo_13473305
Meslo_13474097
Meslo_13474522
Meslo_13475817
Meslo_13475942
Meslo_13474014
Metac_20090100
Metth_15678506
Oceih_23100619
Oceih_23098332
Oniye_39938976
Oniye_39938620
Pasmu_15601990
Pasmu_15603594
Agrtu_15888162
Acine_50085668
Agrtu_16119640
Agrtu_15891435
Agrtu_15887957
Agrtu_15888729
Agrtu_15890417
Agrtu_15890732
Agrtu_15891761
Agrtu_16119689
Agrtu_15890737
Agrtu_15891743
Agrtu_15891779
Agrtu_15891829
Agrtu_16119287
Arcfu_11499199
Bacsu_16080435
Bacsu_16080423
Biflo_23465528
Borbr_33601144
Borbr_33601620
Borbr_33601951
Borbr_33602284
Borbr_33603522
Braja_27378209
Braja_27379091
Braja_27381509
Braja_27382025
Braja_27382710
Braja_27378421
Brume_17988945
Camje_15792248
Cloac_15893668
Cloac_15894272
Cloac_15896853
Corgl_19551867
Coxbu_29653826
Dehet_57234795
Desps_51245412
Meslo_13470759
Meslo_13473259
Meslo_13473305
Meslo_13474097
Meslo_13474522
Meslo_13475817
Meslo_13475942
Meslo_13474014
Metac_20090100
Metth_15678506
Oceih_23100619
Oceih_23098332
Oniye_39938976
Oniye_39938620
Pasmu_15601990
Pasmu_15603594
Agrtu_15888162
Acine_50085668
Agrtu_16119640
Agrtu_15891435
Agrtu_15887957
Agrtu_15888729
Agrtu_15890417
Agrtu_15890732
Agrtu_15891761
Agrtu_16119689
Agrtu_15890737
Agrtu_15891743
Agrtu_15891779
Agrtu_15891829
Agrtu_16119287
Arcfu_11499199
Bacsu_16080435
Bacsu_16080423
Biflo_23465528
Borbr_33601144
Borbr_33601620
Borbr_33601951
Borbr_33602284
Borbr_33603522
Braja_27378209
Braja_27379091
Braja_27381509
Braja_27382025
Braja_27382710
Braja_27378421
Brume_17988945
Camje_15792248

Процедура приведения к форме Смита.

```
> restart;with(LinearAlgebra):

> SMITHform:=proc(A::Matrix(integer))
  local m,n,B,U,V,q,ver,hor,i,j,pos,num,s;

m,n:=Dimension(A);
B:=Matrix(A);
U:=IdentityMatrix(m,compact=false);
V:=IdentityMatrix(n,compact=false);
if Equal(B,ZeroMatrix(m,n)) then
  RETURN(B,U,V,0);
end if;

s:=1; # Счетчик включен.
```

```
LB_nw:
# Перемещение наименьшего
# (по абсолютной величине) элемента
# юго-восточного блока B[s..m,s..n].
# в позицию [s,s] (северо-западный угол блока).

pos:=[s,s];
num:=abs(B[s,s]);
for i from s to m do
  for j from s to n do
    if B[i,j]<>0 and (num=0 or abs(B[i,j])<num) then
      pos:=[i,j];
      num:=abs(B[i,j]);
    end if;
  end do;
end do;
```

```
if pos[1]<>s then
  B:=RowOperation(B,[s,pos[1]]);
  U:=RowOperation(U,[s,pos[1]]);
end if;
if pos[2]<>s then
  B:=ColumnOperation(B,[s,pos[2]]);
  V:=ColumnOperation(V,[s,pos[2]]);
end if;

if B[s,s]<0 then
  B:=RowOperation(B,s,-1);
  U:=RowOperation(U,s,-1);
end if;
```

```
LB_modedg:
# Приведение по модулю ключевого элемента
# (в позиции [s,s]) окаймления (edging)
# вниз и вправо от ключевого элемента.

for i from s+1 to m do
  if B[i,s]<>0 then
    q:=iquo(B[i,s],B[s,s]);
    B:=RowOperation(B,[i,s],-q);
    U:=RowOperation(U,[i,s],-q);
  end if;
end do;
```

```
for j from s+1 to n do
  if B[s,j]<>0 then
    q:=iquo(B[s,j],B[s,s]);
    B:=ColumnOperation(B,[j,s],-q);
    V:=ColumnOperation(V,[j,s],-q);
  end if;
end do;
```

```
LB_nulledg:
# Обнуление окаймления.

pos:=[s,s];
num:=abs(B[s,s]);
while not (Equal(B[s+1..m,s],ZeroVector[column](m-s))
           and
           Equal(B[s,s+1..n],ZeroVector[row](n-s))) do
# Перемещение наименьшего
# (по абсолютной величине) элемента окаймления
# в ключевую позицию [s,s].
```

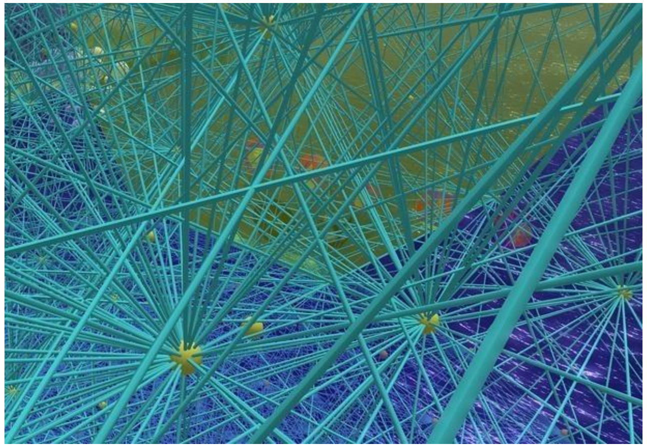
```
for i from s+1 to m do
  if B[i,s]<>0 and abs(B[i,s])<num then
    pos:=[i,s];
    num:=abs(B[i,s]);
  end if;
end do;

for j from s+1 to n do
  if B[s,j]<>0 and abs(B[s,j])<num then
    pos:=[s,j];
    num:=abs(B[s,j]);
  end if;
end do;
```

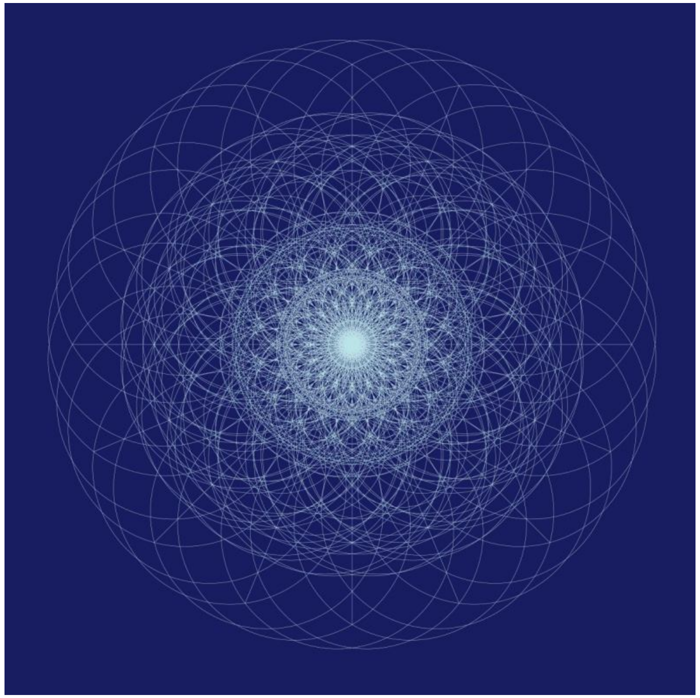
```
  if pos[2]=s and pos[1]>s then
    B:=RowOperation(B,[s,pos[1]]);
    U:=RowOperation(U,[s,pos[1]]);
  elif pos[1]=s and pos[2]>s then
    B:=ColumnOperation(B,[s,pos[2]]);
    V:=ColumnOperation(V,[s,pos[2]]);
  end if;

  goto(LB_modedg);
  # Возврат к приведению по модулю
  # с новым (меньшим по абсолютной величине)
  # ключевым элементом.
end do;
if B[s,s]<0 then
  B:=RowOperation(B,s,-1);
  U:=RowOperation(U,s,-1);
end if;
```

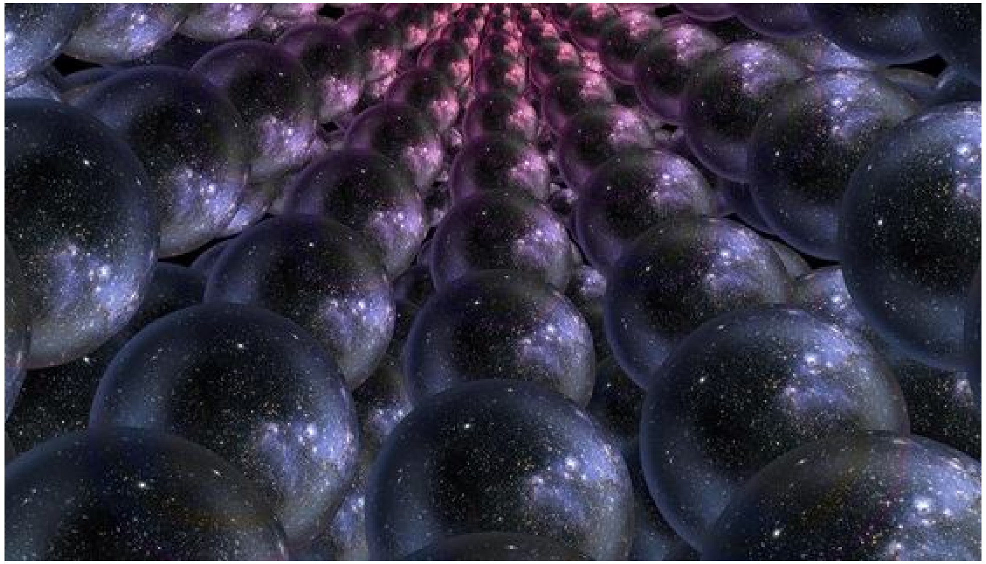
```
# Проверка юго-восточного блока
# (справа снизу от ключевого элемента):
# все элементы должны делиться на ключевой.
pos:=[s,s];
num:=B[s,s];
for i from s+1 to m do
  for j from s+1 to n do
    if B[i,j] mod B[s,s]<>0 and
       B[i,j] mod B[s,s]<num then
      pos:=[i,j];
      num:=B[i,j] mod B[s,s];
    end if;
  end do;
end do;
```



```
if pos<>[s,s] then
  # В случае невыполнения условия делимости
  # "портим" окаймление, прибавляя к текущей
  # (ключевой) строке строку, содержащую наименьший
  # остаток при делении на ключевой элемент.
  B:=RowOperation(B,[s,pos[1]],1);
  U:=RowOperation(U,[s,pos[1]],1);
```



```
# Ключевая строка приводится по модулю
# ключевого элемента; столбец в котором
# располагался элемент с наименьшим остатком,
# меняется местами с ключевым столбцом.
q:=iquo(B[s,pos[2]],B[s,s]);
B:=ColumnOperation(B,[pos[2],s],-q);
V:=ColumnOperation(V,[pos[2],s],-q);
B:=ColumnOperation(B,[pos[2],s]);
V:=ColumnOperation(V,[pos[2],s]);
```



```
  goto(LB_nulledg);
  # Возврат к этапу обнуления окаймления
  # с новым (меньшим по абсолютной величине)
  # ключевым элементом.
end if;

if Equal(B[s+1..m,s+1..n],ZeroMatrix(m-s,n-s)) then
  # Если очередной юго-восточный блок
  # (справа снизу от ключевого элемента)
  # оказывается нулевым (или пустым),
  # то происходит останов.
  # Возвращаются: форма Смита, матрицы перехода и ранг.
  RETURN(B,U,V,s);
else
```

```
    # Если останов не достигнут,
    # то получает приращение счетчик.
    s:=s+1;

    goto(LB_nw);
    # Работа возобновляется с новой ключевой позиции.

end if;

end proc;
```

```
> A:=Matrix([[50, -88, 50, -62], [-156, 264, -150, 186],
                                        [18, -42, 24, -30]]);
```

$$A := \begin{bmatrix} 50 & -88 & 50 & -62 \\ -156 & 264 & -150 & 186 \\ 18 & -42 & 24 & -30 \end{bmatrix}$$

```
> B,U,V,r:=SMITHform(A);U.A.V-B;
```

$$B, U, V, r := \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 6 & 0 & 0 \\ 0 & 0 & 6 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 1 \\ -6 & -2 & 3 \\ -150 & -43 & 44 \end{bmatrix}, \begin{bmatrix} -21 & 65 & -3 & 0 \\ -11 & 34 & -1 & 1 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, 3$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

```
> A:=Matrix([[500,-828,150,-621,122],
  [-156,264,-150,186,231],[188,-428,274,-307,177],
  [198,-438,574,-507,777]]);
```

$$A := \begin{bmatrix} 500 & -828 & 150 & -621 & 122 \\ -156 & 264 & -150 & 186 & 231 \\ 188 & -428 & 274 & -307 & 177 \\ 198 & -438 & 574 & -507 & 777 \end{bmatrix}$$

```
> B:=SMITHform(A)[1];U:=SMITHform(A)[2];V:=SMITHform(A)[3];
```

$$B := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 30 & 0 \end{bmatrix}$$

$U :=$

[307 , 0 , 0 , 1]

[7673120958 , 0 , -5868864 , 25990685]

[885190952604587815 , 992583 , -677047229914751 , 2998352213206080]

[959854419406078321344862 5 , 10763046960365 , -734154335703666543502 8 ,
3251255126720028478257 3]

$$V := \begin{bmatrix} 0 & 344165200255 & 2127302931360097459 & -2942825108782 & -185304363 \\ 0 & 0 & -3362194 & -8994803 & -20067583 \\ 0 & -22636834 & -139919445317013 & 173526336 & -44680878 \\ 1 & 2155625602277 & 13324033514264162570 & -18431472095632 & -137294956 \\ 5 & 9394470687364 & 58067709974754251894 & -80326270939118 & -20670600 \end{bmatrix}$$

Замечания. *(1) Команда* **goto** *является в* **Maple** *"нелегальной", она противоречит идеологии* **Maple***-программирования. В качестве упражнения вам предлагается переделать процедуру* **SMITHform**, *без использования* **goto**.

(2) В пакете **LinearAlgebra** *предусмотрена стандартная версия процедуры приведения к форме Смита – функция* **SmithForm(A)**.

АЛГОРИТМ РЕШЕНИЯ СИСТЕМЫ ЛИНЕЙНЫХ УРАВНЕНИЙ НАД КОЛЬЦОМ $\mathbb{Z}$

(с помощью формы Смита)

Рассмотрим с.л.у.

$$A \cdot x = b; \qquad (1)$$

где

A - целочисленная $(m \times n)$-матрица;
b - целочисленный $(m \times 1)$-вектор,
$\mathbf{x}$ - неизвестный целочисленный $(n \times 1)$-вектор.

[1.] Приведем матрицу $\boldsymbol{A}$ к **нормальной форме Смита**:

$$\boldsymbol{U\cdot A\cdot V = S = \mathrm{diag}(\mu_1, \dots, \mu_r)};$$

$$\boldsymbol{U \in GL(n,\mathbb{Z}); \; V \in GL(n,\mathbb{Z}); \; r = \mathrm{rank}(A)}.$$

[2.] Произведем **замену неизвестного** вектора

$$\boldsymbol{x = V\cdot y} \qquad (2)$$

и домножим обе части уравнения **(1)** на матрицу $\boldsymbol{U}$; получим:

$\boldsymbol{(U\cdot A\cdot V)\cdot y = c}$, где $\boldsymbol{c = U\cdot b}$, или

$$\boldsymbol{S\cdot y = c}. \qquad (3)$$

[3.] Перейдем к подробной записи системы **(3)**:

$$\begin{cases} \boldsymbol{\mu_i\cdot y_i = c_i}; & \boldsymbol{i = 1, \dots, r}; \\ \boldsymbol{0 = c_i}; & \boldsymbol{i = r+1, \dots, m}. \end{cases}$$

Система **(3)** будет **несовместной** (над $\mathbb{Z}$), если хотя бы одно из чисел $\boldsymbol{c_i}$ $(\boldsymbol{i = r+1, \dots, m})$ **отлично от нуля** или хотя бы одно из чисел $\boldsymbol{c_i}$ $(\boldsymbol{i = 1, \dots, r})$ **не делится** на соответствующее число $\boldsymbol{\mu_i}$.

[4.] В противном случае система (3) будет **совместной**. При $r = n$ она будет **определенной**; ее единственное решение определится по формулам:

$$y_i = c_i / \mu_i\,; \quad i = 1, \dots, n.$$

[5.] Если же $r < n$, то (3) окажется **неопределенной**, неизвестные с номерами, большими r, будут **свободными**; общее решение запишется в виде:

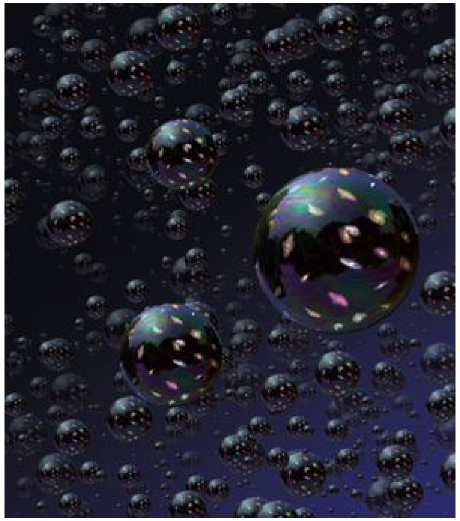
$$y_i = c_i / \mu_i\,; \quad i = 1, \dots, r;$$
$$y_i = y_i; \quad i = r + 1, \dots, n.$$

[6.] Возвращаемся **к исходным неизвестным** по формуле (2).

ПРОЦЕДУРНАЯ РЕАЛИЗАЦИЯ

```
> iLinSolve:=proc(A::Matrix(integer),
                  b::Vector(integer))
  local m,n,U,V,S,r,c,i,y,x;
m,n:=Dimension(A);
if m<>Dimension(b) then
  ERROR();
end if;
S,U,V,r:=SMITHform(A);
print(evaln(S)=S);
```

```
if r=0 and not Equal(b,ZeroVector(m)) then
  RETURN(NULL);
elif r=0 and Equal(b,ZeroVector(m)) then
  RETURN(Vector(n,symbol=x));
else
  c:=U.b;
  print(evaln(c)=c);
  y:=Vector(n,symbol=y);
  print(evaln(y)=y);
  if not Equal(c[r+1..m],ZeroVector(m-r)) then
    print(evaln(y)=NULL);
    RETURN(NULL);
```



```
  else
    for i from 1 to r do
      if c[i] mod S[i,i]<>0 then
        print(evaln(y)=NULL);
        RETURN(NULL);
      else
        y[i]:=c[i]/S[i,i];
      end if;
    end do;
    print(evaln(y)=y);
    x:=V.y;
    print(evaln(x)=x);
    RETURN(x);
  end if;
end if;
end proc;
```

Примеры.

```
> A,b:=Matrix([[0,-14,-2,16],[4,-34,-4,40],
                  [4,2,0,0]]),Vector([8,8,64]);
```

$$A, b := \begin{bmatrix} 0 & -14 & -2 & 16 \\ 4 & -34 & -4 & 40 \\ 4 & 2 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 8 \\ 8 \\ 64 \end{bmatrix}$$

```
> iLinSolve(A,b);
```

$$S = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 8 & 0 \end{bmatrix} \quad c = \begin{bmatrix} -8 \\ 64 \\ 184 \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \quad y = \begin{bmatrix} -4 \\ 32 \\ 23 \\ y_4 \end{bmatrix} \quad x = \begin{bmatrix} y_4 \\ 32 - 2\,y_4 \\ -44 - 2\,y_4 \\ 23 - 2\,y_4 \end{bmatrix}$$

$$\begin{bmatrix} y_4 \\ 32 - 2\,y_4 \\ -44 - 2\,y_4 \\ 23 - 2\,y_4 \end{bmatrix}$$

```
> A,b:=Matrix([[-18,32,40,10,50],[-6,20,10,10,20],
              [-4,10,8,4,12],[-28,62,58,24,62]]),
              Vector([64,34,18,116]);
```

$$A, b := \begin{bmatrix} -18 & 32 & 40 & 10 & 50 \\ -6 & 20 & 10 & 10 & 20 \\ -4 & 10 & 8 & 4 & 12 \\ -28 & 62 & 58 & 24 & 62 \end{bmatrix}, \begin{bmatrix} 64 \\ 34 \\ 18 \\ 116 \end{bmatrix}$$

```
> iLinSolve(A,b);
```

$$S = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 60 & 0 \end{bmatrix} \quad c = \begin{bmatrix} -16 \\ -10 \\ 6 \\ 60 \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} \quad y = \begin{bmatrix} -8 \\ -5 \\ 3 \\ 1 \\ y_5 \end{bmatrix} \quad x = \begin{bmatrix} -4 + 5\, y_5 \\ 1 \\ -1 + 2\, y_5 \\ y_5 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} -4 + 5\, y_5 \\ 1 \\ -1 + 2\, y_5 \\ y_5 \\ 0 \end{bmatrix}$$

Matriz de halloween