КАК КОМПЬЮТЕР ПОМОГАЕТ УПРОЩАТЬ АЛГЕБРАИЧЕСКИЕ УРАВНЕНИЯ

Блинков Юрий Анатольевич

Российский Университет Дружбы Народов (РУДН)
Руководитель научного центра вычислительных методов в прикладной математике

«Саратовский национальный исследовательский
государственный университет имени Н.Г. Чернышевского»
зав. кафедрой «математического и компьютерного моделирования»

11 марта 2021

Все исходные файлы доклада https://github.com/blinkovua/Smart-Week

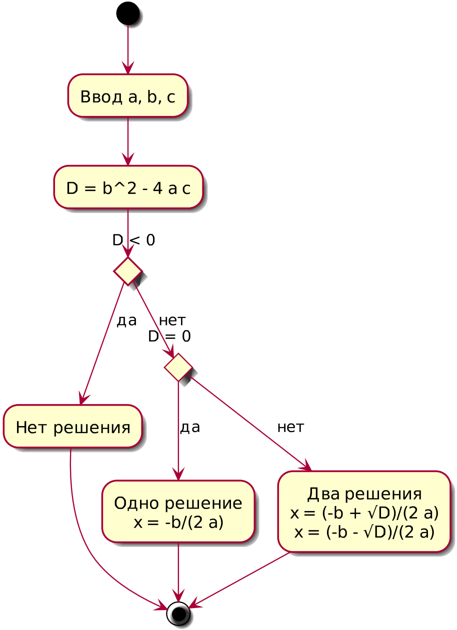
Дополнительные материалы

Васильев Н. Н. КАК КОМПЬЮТЕР ПОМОГАЕТ УПРОЩАТЬ АЛГЕБРАИЧЕСКИЕ УРАВНЕНИЯ. Компьютерные инструменты в образовании.
http://cte.eltech.ru/ojs/index.php/kio/article/view/964

Васильев Н. Н. УПРОЩЕНИЕ СИСТЕМЫ ПОЛИНОМИАЛЬНЫХ УРАВНЕНИЙ. Компьютерные инструменты в образовании.
http://cte.eltech.ru/ojs/index.php/kio/article/view/970

5 января 2021 г. ушел из жизни Владимир Петрович Гердт, доктор физико-математических наук, профессор, начальник сектора алгебраических и квантовых вычислений Научного отдела вычислительной физики Лаборатории информационных технологий Объединенного института ядерных исследований, г. Дубна.

```
(*) --> "Ввод a, b, c"
"Ввод a, b, c" --> "D = b^2 - 4 a c"
if "D < 0" then
  -->[да] "Нет решения"
  --> (*)
else
  -->[нет] if "D = 0" then
             -->[да] "Одно решение\nx = -b/(2 a)"
             --> (*)
           else
             -->[нет] "Два решения\nx = (-b + D)/(2 a)\nx = (-b - D)/(2 a)"
             --> (*)
           endif
endif
```


Ввод a, b, c
D = b^2 - 4 a c
D < 0
да
нет
D = 0
Нет решения
да
нет
Одно решение
x = -b/(2 a)
Два решения
x = (-b + √D)/(2 a)
x = (-b - √D)/(2 a)

Graphviz (сокращение от англ. Graph Visualization Software)

пакет утилит по автоматической визуализации графов, заданных в виде описания на языке DOT, а также дополнительных TUI и GUI программ, виджетов и библиотек, используемых при разработке программного обеспечения для визуализации структурированных данных. Пакет Graphviz разработан специалистами лаборатории AT&T и распространяется с открытыми исходными файлами по лицензии EPL (Eclipse Public License) и работает на многих операционных системах, включая Linux, Mac OS, Unix-подобные, Microsoft Windows.

Сайт graphviz.org

In [2]:
```
x, y, z = sympy.symbols('x, y, z')
```

In [3]:
```
eq = 3*x**2 - 4*x + 1
eq
```

Out[3]: $3x^2 - 4x + 1$

In [4]:
```
s = sympy.solve(eq, x)
display(s[0])
display(s[1])
s
```

$$\frac{1}{3}$$

$$1$$

Out[4]: [1/3, 1]

In [5]:
```
eq = 3*x**2 + 2*x + 1
eq
```

Out[5]: $3x^2 + 2x + 1$

In [6]:
```
s = sympy.solve(eq, x)
display(s[0])
display(s[1])
s
```

$$-\frac{1}{3} - \frac{\sqrt{2}i}{3}$$

$$-\frac{1}{3} + \frac{\sqrt{2}i}{3}$$

Out[6]:
```
[-1/3 - sqrt(2)*I/3, -1/3 + sqrt(2)*I/3]
```

In [7]:
```
eq = x**2 + 2*x + 1
eq
```
Out[7]: $x^2 + 2x + 1$

In [8]:
```
s = sympy.solve(eq, x)
display(s[0])
s
```
-1

Out[8]: [-1]

```
In [22]: eq1 = x**2 + x - 2
         eq2 = x**2 - 4*x + 3

In [23]: eq1 - eq2
```

Out[23]: $5x - 5$

```
In [24]: eq1 = (eq1 - eq2)/5
         eq1
```

Out[24]: $x - 1$

In [25]:
```
(eq2 - x*eq1).expand()
```
Out[25]: $3 - 3x$

In [26]:
```
eq2 = ((eq2 - x*eq1)/3).expand()
```

In [28]:
```
eq2 + eq1
```
Out[28]: 0

In [29]:
```
eq2 = eq2 + eq1
```

In [30]:
```
eq1, eq2
```
Out[30]:
```
(x - 1, 0)
```

In [17]:
```python
eq1 = x**2 + x - 2
eq2 = x**2 - 4*x - 3
```

In [18]:
```python
eq1 - eq2
```

Out[18]: $5x + 1$

In [19]:
```python
eq1 = eq1 - eq2
eq1
```

Out[19]: $5x + 1$

In [20]:
```
(5*eq2 - x*eq1).expand()
```
Out[20]: $-21x - 15$

In [21]:
```
eq2 = (5*eq2 - x*eq1).expand()
eq2
```
Out[21]: $-21x - 15$

In [22]:
```
5*eq2 + 21*eq1
```
Out[22]: -54

In [23]:
```
eq1 = 3*x + 5*y - 1
eq2 = x + 3*z  + 3
```

In [24]:
```
(eq1 - 3*eq2).expand()
```

Out[24]: $5y - 9z - 10$

In [25]:
```
eq1 = (eq1 - 3*eq2).expand()
eq1
```

Out[25]: $5y - 9z - 10$

In [26]:
```
eq1, eq2
```

Out[26]:
```
(5*y - 9*z - 10, x + 3*z + 3)
```

Определение

(Допустимым) мономиальным упорядочением на множестве мономов называется линейный порядок, удовлетворяющий следующим свойствам:

- для любого момома a выполнено $1 \prec a$;
- если $a \prec b$, то для любого c выполнено $a \cdot c \prec b \cdot c$.

Случай одной переменной

Можно задать только один допустимый порядок — по степени переменной:

$$x \prec x^2 \prec x^3 \prec x^4 \prec x^5 \prec x^6 \prec \dots$$

Случай нескольких переменных

Сначала задают порядок переменных, например: $z \prec y \prec x$
Наиболее часто используют *lex* и *deglex*.
lex: $z \prec z^2 \prec \dots \prec y \prec y \cdot z \prec y \cdot z^2 \prec \dots \prec y^2 \prec y^2 \cdot z \prec y^2 \cdot z^2 \prec \dots \prec x \prec x \cdot z \prec x \cdot z^2 \dots\dots \prec x \cdot y \prec x \cdot y \cdot z \prec x \cdot y \cdot z^2 \prec \dots$
deglex: $z \prec y \prec x \prec z^2 \prec y \cdot z \prec y^2 \prec x \cdot z \prec x \cdot y \prec x^2 \prec z^3 \prec \dots$

In [27]:
```
variables = sympy.symbols('x, y, z')
Monom.variables = len(variables) + 1
# POTlex TOPlex POTdeglex TOPdeglex
Monom.cmp = Monom.POTlex
Monom.zero = Monom(0 for v in range(Monom.variables))
for i in range(len(variables)):
    p = Poly()
    p.append([Monom(0 if l-1 != i else 1 for l in range(Monom.variables)), 1])
    globals()[str(variables[i])] = p
```

In [28]:
```
def tosympy(p):
    r = 0
    for m, c in p:
        for i in range(1, len(m)):
            c *= variables[i-1]**m[i]
        r += c
    return r
```

In [29]:
```
x
```

Out[29]:
```
[[(0, 1, 0, 0), 1]]
```

In [30]:
```
tosympy(x)
```

Out[30]: x

lex

```
In [31]: (x**2 + y + 1)**3
```

```
Out[31]: [[(0, 6, 0, 0), 1],
          [(0, 4, 1, 0), 3],
          [(0, 4, 0, 0), 3],
          [(0, 2, 2, 0), 3],
          [(0, 2, 1, 0), 6],
          [(0, 2, 0, 0), 3],
          [(0, 0, 3, 0), 1],
          [(0, 0, 2, 0), 3],
          [(0, 0, 1, 0), 3],
          [(0, 0, 0, 0), 1]]
```

```
In [32]: tosympy((x**2 + y + 1)**3)
```

Out[32]: $x^6 + 3x^4y + 3x^4 + 3x^2y^2 + 6x^2y + 3x^2 + y^3 + 3y^2 + 3y + 1$

deglex

In [34]:
```
(x**2 + y + 1)**3
```

Out[34]:
```
[[(0, 6, 0, 0), 1],
 [(0, 4, 1, 0), 3],
 [(0, 4, 0, 0), 3],
 [(0, 2, 2, 0), 3],
 [(0, 2, 1, 0), 6],
 [(0, 0, 3, 0), 1],
 [(0, 2, 0, 0), 3],
 [(0, 0, 2, 0), 3],
 [(0, 0, 1, 0), 3],
 [(0, 0, 0, 0), 1]]
```

In [35]:
```
tosympy((x**2 + y + 1)**3)
```

Out[35]: $x^6 + 3x^4y + 3x^4 + 3x^2y^2 + 6x^2y + 3x^2 + y^3 + 3y^2 + 3y + 1$

deglex

In [72]:
```
G = GB()
G.algorithm1([x**3 + y**2 + z - 1,
              y**3 + z**2 + x - 1,
              z**3 + x**2 + y - 1])
for g in G:
    print(g)
for g in G:
    display(tosympy(g))
print("time %.2f" % G.time)
```

```
[[(0, 0, 0, 3), 1], [(0, 2, 0, 0), 1], [(0, 0, 1, 0), 1], [
[[(0, 0, 3, 0), 1], [(0, 0, 0, 2), 1], [(0, 1, 0, 0), 1], [
[[(0, 3, 0, 0), 1], [(0, 0, 2, 0), 1], [(0, 0, 0, 1), 1], [
```

$$x^2 + y + z^3 - 1$$

$$x + y^3 + z^2 - 1$$

$$x^3 + y^2 + z - 1$$

```
time 0.00
```

lex

In [74]:
```
G = GB()
G.algorithm1([x**3 + y**2 + z - 1,
              y**3 + z**2 + x - 1,
              z**3 + x**2 + y - 1])
for g in G:
    display(tosympy(g))
print("time %.2f" % G.time)
```

$$-z^{26} + 9z^{23} - 29z^{20} + 6z^{18} + 11z^{17} + 14z^{16} - 27z^{15} + 110z^{14} - 37z^{13} - 4z^{12} - 163z^{11} - 36z^{10} + 173z^{9} - 28z^{8} + 146z^{7} - 208z^{6} + 94z^{5} - 89z^{4} + 91z^{3} - 37z^{2} + 5z$$

$$4468898502239659 62yz - 1485873290608935 51z^{25} + 6922726552813468 3z^{24} + 17710710631862173 1z^{23} + 15701335130592671 76z^{22} - 34503225368991417 4z^{21} - 12769185149777700 00z^{20} - 60532612817901497 27z^{19} - 96270871433086167z^{18} + 36128062885118090 63z^{17} + 52891289817083087 90z^{16} + 52840587681667448 10z^{15} - 24735137624311013 45z^{14} + 18022223916861656497z^{13} - 9199826957057177674z^{12}$$

lex

$$
\begin{aligned}
&65719095621171465x + 65719095621171465y + 7467678205870943z^{25}\\
&+ 28598228905708245z^{24} + 40958640195518500z^{23} - 18118488190106417z^{22}\\
&- 20098786645586579 0z^{21} - 30335051628035530 0z^{20}\\
&- 15240600673816274 8z^{19} + 40312229419945126 0z^{18}\\
&+ 64922637468701860 7z^{17} + 61815056533758500 2z^{16}\\
&+ 35303454987842063 3z^{15} + 23880255957127360 6z^{14}\\
&- 17482726389772882 5z^{13} - 19729281848006965 99z^{12}\\
&- 23278200587814248 73z^{11} - 13459706427049345 06z^{10}\\
&+ 20878614215277882 38z^{9} + 31173718178910977 26z^{8}\\
&+ 11223979673447842 74z^{7} - 43789036272389351 8z^{6}\\
&- 20021241591089244 66z^{5} - 42413702095534382z^{4} - 84397489592758113z^{3}\\
&+ 70261542614873087 2z^{2} - 24065368690169090 4z - 65719095621171465
\end{aligned}
$$

`time 8.34`

lex

```
In [75]: sympy.count_roots(tosympy(G[0]))
Out[75]: 6
```

```
In [76]: sympy.intervals(tosympy(G[0]))
Out[76]: [((0, 0), 1),
          ((0, 1/3), 1),
          ((1/3, 1/2), 1),
          ((1/2, 2/3), 1),
          ((2/3, 1), 1),
          ((1, 1), 1)]
```

```
In [77]: sympy.nroots(tosympy(G[0]))
Out[77]: [0,
          0.329186259200067,
          0.450908604847272,
          0.543689012692076,
          0.761009624180514,
          1.00000000000000,
          -1.2150637681837 - 0.125105344727484*I,
```

lex: старшие члены z^{27}, y, x

мономов без делителя на старшие члены $= 27$

$$1_{\mathbb{M}}, z, z^2, \ldots, z^{26}$$

deglex: старшие члены z^3, y^3, x^3

мономов без делителя на старшие члены $= 27$

$$1_{\mathbb{M}}, = 1$$

$$z, y, x, = 3$$
$$z^2, yz, y^2, xz, xy, x^2, = 6$$
$$yz^2, y^2z, xz^2, xyz, xy^2, x^2z, x^2y, = 7$$
$$y^2z^2, xyz^2, xy^2z, x^2z^2, x^2yz, x^2y^2, = 6$$
$$xy^2z^2, x^2yz^2, x^2y^2z, = 3$$
$$x^2y^2z^2. = 1$$

jupyter monom.py✔ 11/06/2018

File Edit View Language

```python
# -*- coding: utf-8 -*-

class Monom(tuple):
  variables = None
  zero = None

  def __new__ (cls, *args):
    if len(args) == 0:
      return Monom.zero
    else:
      return super().__new__(cls, *args)

  def __init__(self, *args):
    assert len(self) == Monom.variables
    assert all(v >= 0 for v in self)

  def position(self):
    return self[0]

  def degree(self):
    return sum(self[1:])
```

jupyter poly.py 06.11.2018

File Edit View Language

```python
# -*- coding: utf-8 -*-

import sympy
from sympy import S, Basic, gcd
from monom import *

class Poly(list):
  def __init__(self):
    super().__init__()

  def copy(self):
    p = Poly()
    for m, k in self:
      p.append([m, k])
    return p

  def lm(self):
    assert self
    return self[0][0]
```

базис Грёбнера, алгоритм Бухбергера, инволютивный

jupyter gb.py✔ 07.11.2018

File Edit View Language

```python
# -*- coding: utf-8 -*-

import time

from poly import *

class GB(list):
  def __init__(self):
    super().__init__()

  def __str__(self):
    return ",\n".join(str(p) for p in self)

  def __repr__(self):
    return ",\n".join(repr(p) for p in self)

  def __addS(self, p1, p2):
    if  p1.lm().position() == p2.lm().position():
      m = p1.lm().lcm(p2.lm())
      if m.degree() == p1.lm().degree() + p2.lm().degree():
        self.crit1 += 1
```

https://www.sympy.org/en/index.html

Структура университета » Мехмат »

Кафедра математического и компьютерного моделирования

Адрес:
410012, г. Саратов, ул. Астраханская, 83, корпус 9, ком. 301, 302
Телефон: +7 (8452) 51 - 84 - 80
Email:
mexmat_MKM@info.sgu.ru

Все исходные файлы доклада https://github.com/blinkovua/Smart-Week

Допольнительные материалы

Васильев Н. Н. КАК КОМПЬЮТЕР ПОМОГАЕТ УПРОЩАТЬ АЛГЕБРАИЧЕСКИЕ УРАВНЕНИЯ. Компьютерные инструменты в образовании.
http://cte.eltech.ru/ojs/index.php/kio/article/view/964

Васильев Н. Н. УПРОЩЕНИЕ СИСТЕМЫ ПОЛИНОМИАЛЬНЫХ УРАВНЕНИЙ. Компьютерные инструменты в образовании.
http://cte.eltech.ru/ojs/index.php/kio/article/view/970