Wykład 10

Elementy programowania: wzorce: _, x_, x:pattern, _/;,równoważność struktur, x_:X, x_.

In[1]:=
```
ClearAll["Global`*"]
```
wyczyść wszystko

Przypomnienie: Map, MapAt

In[2]:=
```
Map[f, {a, {b, G}, c}]
```
zastosuj do

Out[2]= $\{f(a), f(\{b, G\}), f(c)\}$

In[3]:=
```
f /@ {a, {b, G}, c}
```

Out[3]= $\{f(a), f(\{b, G\}), f(c)\}$

In[4]:=
```
MapAll[f, {a, {b, G}, c}]
```
zastosuj do wszystkich

Out[4]= $f(\{f(a), f(\{f(b), f(G)\}), f(c)\})$

In[5]:=
```
f //@ {a, {b, G}, c}
```

Out[5]= $f(\{f(a), f(\{f(b), f(G)\}), f(c)\})$

In[6]:=
```
f /@ h[p[a, b, c], d[g, f, h]]
```

Out[6]= $h(f(p(a, b, c)), f(d(g, f, h)))$

In[7]:=
```
f //@ h[p[a, b, c], d[g, f, h]]
```

Out[7]= $f(h(f(p(f(a), f(b), f(c))), f(d(f(g), f(f), f(h)))))$

In[8]:=
```
Map[f, h[p[a, b, c], d[g, f, h]], {2}] (* na drugim poziomie *)
```
zastosuj do

Out[8]= $h(p(f(a), f(b), f(c)), d(f(g), f(f), f(h)))$

Jeśli ustawimy Heads -> True

In[9]:=
```
Map[FF, h[p[a, b, c], d[g, f, h]], {2}, Heads -> True]
```
zastosuj do | głowy | prawda

Out[9]= $h(\mathrm{FF}(p)(\mathrm{FF}(a), \mathrm{FF}(b), \mathrm{FF}(c)), \mathrm{FF}(d)(\mathrm{FF}(g), \mathrm{FF}(f), \mathrm{FF}(h)))$

Ciekawe zastosowanie:

In[10]:=
```
mojewyrazenie =.
mojewyrazenie = a + b / c + (d + e / f) / (r + s / t)
```

Out[11]= $a + \frac{b}{c} + \frac{d + \frac{e}{f}}{r + \frac{s}{t}}$

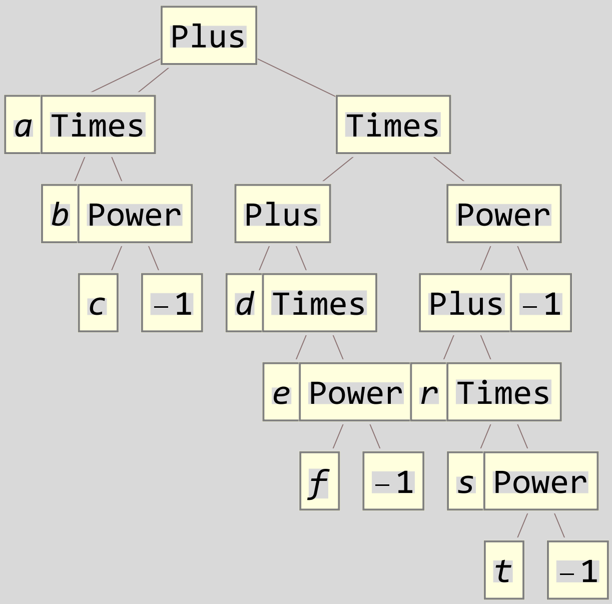
In[12]:=
```
mojewyrazenie
% // FullForm
```
pełna forma
```
%% // TreeForm
```
w formie drz

Out[12]= $a + \frac{b}{c} + \frac{d + \frac{e}{f}}{r + \frac{s}{t}}$

Out[13]//FullForm=

Plus[a, Times[b, Power[c, −1]],
  Times[Plus[d, Times[e, Power[f, −1]]], Power[Plus[r, Times[s, Power[t, −1]]], −1]]]

Out[14]//TreeForm=



In[15]:=
```
poz = Position[mojewyrazenie, Power, {3}]
```
pozycja potęga

Out[15]= $\begin{pmatrix} 2 & 2 & 0 \\ 3 & 2 & 0 \end{pmatrix}$

In[16]:=
```
Part[mojewyrazenie, 2, 2, 0]
```
część

Out[16]= Power

In[17]:=
```
Part[mojewyrazenie, 2, 2, 1]
```
część

Out[17]= c

In[18]:=
```
Part[mojewyrazenie, 3, 2, 0]
```
część

Out[18]= Power

In[19]:=
```
Part[mojewyrazenie, 3, 2, 1]
```
część

Out[19]= $r + \frac{s}{t}$

Zastępujemy 0 przez 1 w poz

In[20]:=
```
poz /. (0 → 1)
```

Out[20]= $\begin{pmatrix} 2 & 2 & 1 \\ 3 & 2 & 1 \end{pmatrix}$

In[21]:=
```
MapAt @@ {FF, mojewyrazenie, poz /. (0 → 1)}
```
zastosuj w danym miejscu

Out[21]= $a + \frac{b}{\mathrm{FF}(c)} + \frac{d + \frac{e}{f}}{\mathrm{FF}\left(r + \frac{s}{t}\right)}$

In[22]:=
```
MapAt @@ {# /. Power → FFF &, mojewyrazenie, poz}
```
zastosuj w danym··· potęga

Out[22]= $b\,\mathrm{FFF}(c, -1) + \left(d + \frac{e}{f}\right)\mathrm{FFF}\left(r + \frac{s}{t}, -1\right) + a$

In[23]:=
```
MapAt @@ {FF, mojewyrazenie, poz}
```
zastosuj w danym miejscu

Out[23]= $b\,\mathrm{FF}(\mathrm{Power})(c, -1) + \left(d + \frac{e}{f}\right)\mathrm{FF}(\mathrm{Power})\left(r + \frac{s}{t}, -1\right) + a$

Wzorce

_

In[24]:=
```
_(* dowolne wyrażenie ale niepuste!  *)
```

Out[24]= _

In[25]:=
```
_ // FullForm
```
pełna forma

Out[25]//FullForm=
Blank[]

In[26]:=
```
Blank[]
```
puste miejsce

Out[26]=
_

In[27]:=
```
{h[], h[x^2], h[Sin[x]], h[5 + 6 I], g[x], h[b, c], h[{b, c}]} /. h[_] → "Pasuję"
```
sinus jedność urojona

Out[27]=
{$h()$, Pasuję, Pasuję, Pasuję, $g(x)$, $h(b, c)$, Pasuję}

In[28]:=
```
{h[], h[x^2], h[Sin[x]], h[5 + 6 I], g[x], h[b, c], h[{b, c}]} /. h[_, _] → "Pasuję"
```
sinus jedność urojona

Out[28]=
$\{h(), h(x^2), h(\sin(x)), h(5+6\,i), g(x)$, Pasuję, $h(\{b, c\})\}$

Podaj pozycję wszystkich funkcji zależnych od dwóch dowolnych argumentów

In[29]:=
```
Position[{{}, {a}, {b, c}, {c, {d, f}}}, {_, _}]
```
pozycja

Out[29]=
{{3}, {4, 2}, {4}}

In[30]:=
```
{3, x, a^4, Sqrt[1 + x]} // FullForm
```
pierwiastek kwadr⋯ pełna forma
```
_^_ // FullForm
```
pełna forma
```
Cases[{3, x, a^4, Sqrt[1 + x]}, _^_]
```
przypadki pierwiastek kwadratowy

Out[30]//FullForm=
List[3, x, Power[a, 4], Power[Plus[1, x], Rational[1, 2]]]

Out[31]//FullForm=
Power[Blank[], Blank[]]

Out[32]=
$\left\{a^4, \sqrt{x+1}\right\}$

In[33]:=
```
Sqrt[2] // FullForm
```
pierwiastek⋯ pełna forma

Out[33]//FullForm=
Power[2, Rational[1, 2]]

In[34]:=
```
Plus @@ Cases[3 + x + a^4 + Sqrt[1 + x], _^_]
```
suma przypadki pierwiastek kwadratowy

Out[34]=
$a^4 + \sqrt{x+1}$

In[35]:=
```
Plus @@ Cases[3 + x + a^4 + Sqrt[1 + x] + x^3, _^_]
suma    przypadki    pierwiastek kwadratowy
```

Out[35]= $a^4 + x^3 + \sqrt{x+1}$

In[36]:=
```
_^2 // FullForm
       pełna forma
```

Out[36]//FullForm= Power[Blank[], 2]

In[37]:=
```
Cases[3 + x + a^4 + Sqrt[1 + x] + x^3, _^3]
przypadki    pierwiastek kwadratowy
```

Out[37]= $\{x^3\}$

In[38]:=
```
Plus @@ DeleteCases[3 + x + a^4 + Sqrt[1 + x], _^_]
suma    usuń nawiasy    pierwiastek kwadratowy
```

Out[38]= $x + 3$

In[39]:=
```
x_ (*  nazywa każde wyrażenie x z prawej strony definicji albo transformacji *)
```

Out[39]= x_

In[40]:=
```
Pattern[x, Blank[]]
szablon    puste miejsc
```

Out[40]= x_

In[41]:=
```
(h[a] + h[b, c] + h[a, a]) h[d, e, f] /. h[x_, y_] → x^y
```

Out[41]= $(a^a + h(a) + b^c)\, h(d, e, f)$

In[42]:=
```
(h[a] + h[b, c] + h[a, a]) h[d, e, f] /. h[x_, x_] → x^x
```

Out[42]= $(h(b, c) + a^a + h(a))\, h(d, e, f)$

In[43]:=
```
F[x_, x_] := x^2
```

In[44]:=
```
F[2, 4]
```

Out[44]= $F(2, 4)$

In[45]:=
```
F[3, 3]
```

Out[45]= 9

In[46]:=
```
Wykladnik =.
Wykladnik[x_^n_] := n
```

In[48]:=
```
Wykladnik /@ {3, x, a^4, Sqrt[1 + x]}
                         pierwiastek kwa
```

Out[48]= $\left\{\text{Wykladnik}(3),\ \text{Wykladnik}(x),\ 4,\ \frac{1}{2}\right\}$

x : pattern

In[49]:=
```
x : pattern (* oznacza dowolne wyrazenie nazwane x,
 wyraznie x pojawia sie po prawej stronie definicji, przypisania *)
```

Out[49]= x : pattern

h : *obj*

represents the pattern object *obj*, assigned the name h.

In[50]:=
```
Sin[1 + y^2] /. h : Sin[x_ + y_] → {h, x, y}
sinus               sinus
```

Out[50]= $\{\sin(y^2+1),\ 1,\ y^2\}$

In[51]:=
```
{Sin[1 + y^2], Cos[x], Sin[y], Sin[1 + y], Sin[1 + y + y^2]} /.
 sinus          cosinus sinus   sinus       sinus
 h : Sin[x_ + y_] → {h, x, y}
     sinus
```

Out[51]= $\{\{\sin(y^2+1),\ 1,\ y^2\},\ \cos(x),\ \sin(y),\ \{\sin(y+1),\ 1,\ y\},\ \{\sin(y^2+y+1),\ 1,\ y^2+y\}\}$

Zobacz na wynik czwartego i piątego podstawienia!

In[52]:=
```
{Sin[a], Sin[a]} /. {h_, h : Sin[x_]} → h
 sinus   sinus            sinus
```

Out[52]= $\sin(a)$

Zadziała tylko wtedy gdy pierwszy element będzie taki sam jak drugi, który wynosi Sin[x_]

In[53]:=
```
{Cos[a], Sin[a]} /. {h_, h : Sin[x_]} → h
 cosinus sinus            sinus
```

Out[53]= $\{\cos(a),\ \sin(a)\}$

Jaka jest różnica względem poprzedniego przykładu

In[54]:=
```
{Sin[a], Sin[a]} /. {h_, h : Sin[x_]} → h^2
 sinus   sinus            sinus
```

Out[54]= $\sin^2(a)$

In[55]:=
```
{x, Sin[a]} /. {h_, h : Sin[x_]} → h^2
```
Out[55]= $\{x, \sin(a)\}$

_Typ

In[56]:=
```
_h (* oznacza każde wyrażenie z główą (head) h *)
```
Out[56]= _h

In[57]:=
```
Sin[x] + Cos[y^2 Sin[z]] /. _Sin → "Graczyk"
```
Out[57]= $\cos(\text{Graczyk}\, y^2) + \text{Graczyk}$

In[58]:=
```
Sin[x] + Cos[y^2 Sin[z]] /. _Integer → "Graczyk"
```
Out[58]= $\cos(y^{\text{Graczyk}} \sin(z)) + \sin(x)$

In[59]:=
```
F =.
F[x_Sin] := x
```

In[61]:=
```
F[Sin[1]]
F[Sin[x]]
F[Sin[π]]
```
Out[61]= $\sin(1)$

Out[62]= $\sin(x)$

Out[63]= $F(0)$

In[64]:=
```
Head[Sin]
```
Out[64]= Symbol

In[65]:=
```
Sin[x] + Cos[y^2 Sin[z]] // FullForm
```
Out[65]//FullForm= Plus[Cos[Times[Power[y, 2], Sin[z]]], Sin[x]]

In[66]:=
```
Sin[x] + Cos[y^2 Sin[z]] /. _Symbol → "Graczyk"
```
sinus cosinus sinus

Out[66]= Graczyk(Graczyk(Graczyk(Graczyk(Graczyk, 2), Graczyk(Graczyk))), Graczyk(Graczyk))

In[67]:=
```
Expand[(x + y + z)^3]
```
rozszerz

Out[67]= $x^3+3x^2y+3x^2z+3xy^2+6xyz+3xz^2+y^3+3y^2z+3yz^2+z^3$

In[68]:=
```
Cases[%, _Symbol, Infinity]
% // Union
```
przypadki nieskończono
suma zbiorów

Out[68]= $\{x, x, y, x, y, y, x, z, x, y, z, y, z, x, z, y, z, z\}$

Out[69]= $\{x, y, z\}$

Średnia geometryczna

In[70]:=
```
geomen[x_List] := Apply[Times, x]^(1/Length[x])
```
zasto··· mnożenie długość

In[71]:=
```
geomen[{a, b, c, d, f}]
```

Out[71]= $\sqrt[5]{a\,b\,c\,d\,f}$

In[72]:=
```
geomen2[x_List] := (Times @@ x)^(1/Length[x])
```
mnożenie długość

In[73]:=
```
geomen2[{a, b, c, d}]
```

Out[73]= $\sqrt[4]{a\,b\,c\,d}$

In[74]:=
```
geomen3[x_List] := Power @@ {(Times @@ x), (1/Length[x])}
```
potęga mnożenie długość

In[75]:=
```
geomen3[{a, b, c, d, e}]
```

Out[75]= $\sqrt[5]{a\,b\,c\,d\,e}$

Testy

?

Możemy ograniczyć dany wzorzec poprzez wprowadzenie testu

In[76]:=
```
wozrzec?test
```
Out[76]=
```
wozrzec?test
```

In[77]:=
```
Clear[F]
```
wyczyść

In[78]:=
```
F[n_Integer?OddQ] := -1
F[n_Integer?EvenQ] := 1
```
wprowadź ⋯ | liczba nieparzys
wprowadź ⋯ | liczba parzysta?

In[80]:=
```
Array[F, {10}]
```
tablica wielowymiarowa

Out[80]= $\{-1, 1, -1, 1, -1, 1, -1, 1, -1, 1\}$

In[81]:=
```
RandomInteger[100, 10]
Count[%, _?EvenQ]
```
losowa liczba całkowita
policz | liczba parzysta?

Out[81]= {35, 35, 49, 88, 42, 31, 32, 77, 11, 100}

Out[82]= 4

Zdefiniujmy wielomian Legendre'a, który zależy od całkowitego dodatniego n i dowolnego x

In[83]:=
```
{NonNegative[-1], NonNegative[1]}
```
nieujemny | nieujemny

Out[83]= {False, True}

In[84]:=
```
Clear[P]
P[n_Integer?NonNegative, x_] := (1/(2^n n!)) D[(x^2-1)^n, {x, n}] // Simplify
```
wyczyść
wprowadź ⋯ | nieujemny | oblicz pochodną | uprość

In[86]:=
```
P[2, x]
```
Out[86]= $\frac{1}{2}\left(3x^2-1\right)$

In[87]:=
```
P[-2, x]
```
Out[87]= $P(-2, x)$

/;

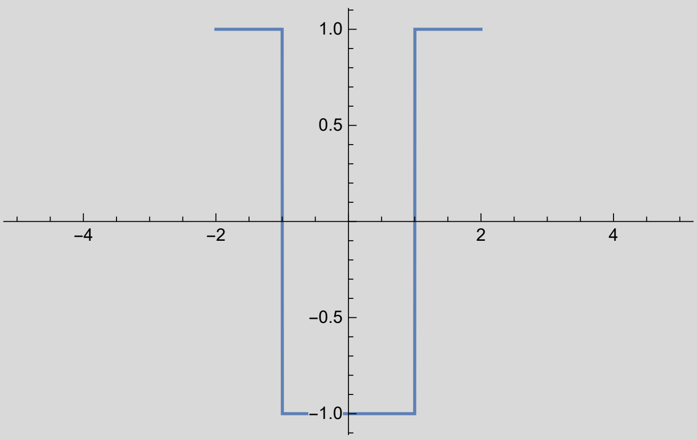
In[88]:=
```
wzorzec /; condition
```
Out[88]= wzorzec /; condition

znaczy dla każdego wzorca takiego, że (jeśli) spełniony jest warunek

In[89]:=
```
Clear[F]
F[x_ /; 4 > x^2 > 1] := 1
F[x_ /; x^2 ≤ 1] := -1
```
wyczyść

In[92]:=
```
Plot[F[x], {x, -5, 5}]
```
wykres

Out[92]=



UWAGA: powyższy przykład zadziała dla obiektów typu Integer, Rational ale nie dla symboli. Dlatego czasem lepiej użyć funkcji

In[93]:=
```
F2[x_] := Piecewise[{{1, x^2 <= 1}, {-1, 4 > x^2 > 1}}]
```
funkcja odcinkowa

In[94]:=
```
F2[x]
```
Out[94]= $\begin{cases} 1 & x^2 \leq 1 \\ -1 & 4 > x^2 > 1 \end{cases}$

In[95]:=
```
F2[0]
```
Out[95]= 1

In[96]:=
```
F[0]
```
Out[96]= -1

In[97]:=
```
Integrate[#, {x, 0, 2}] & /@ {F[x], F2[x]}
```
całka

Out[97]= $\left\{\int_0^2 F(x)\, dx,\ 0\right\}$

In[98]:=
```
? VectorQ
```
VectorQ[*expr*] gives True if *expr* is a list or a one-dimensional SparseArray object, none of whose elements are themselves lists, and gives False otherwise.

VectorQ[*expr*, *test*] gives True only if *test* yields True when applied to each of the elements in *expr*. >>

In[99]:=
```
? NumberQ
```
NumberQ[*expr*] gives True if *expr* is a number, and False otherwise. >>

In[100]:=
```
Cases[{{a, {a, b}}, {a, {b, c}}, 3, {3, 4}, {5, {6, 7}}}, x_ /; VectorQ[x, NumberQ]]
```
przypadki wektor? liczba?

Out[100]= (3 4)

In[101]:=
```
Cases[{{a, {a, b}}, {a, {b, c}}, 3, {3, 4}, {5, {6, 7}}},
  x_ /; VectorQ[x, NumberQ], Infinity] // StandardForm
```
przypadki wektor? liczba? nieskończoność forma standardowa

Out[101]//StandardForm=
```
{{3, 4}, {6, 7}}
```

In[102]:=
```
&& (* operator koniunkcji *)
```

In[102]:=
```
RandomInteger[100, 100]
Cases[%, x_ /; (x > 36 && x < 40)]
```
losowa liczba całkowita

przypadki

Out[102]= {89, 20, 23, 65, 27, 64, 76, 26, 46, 48, 81, 80, 71, 43, 37, 46, 100, 47, 22, 74, 97, 96, 16, 77, 51, 38, 68, 47, 37, 38, 61, 80, 86, 26, 16, 89, 52, 54, 4, 96, 8, 36, 70, 4, 44, 8, 46, 76, 65, 5, 1, 63, 28, 38, 17, 57, 78, 23, 11, 29, 76, 71, 59, 16, 54, 42, 35, 18, 90, 72, 76, 7, 44, 19, 95, 86, 97, 87, 20, 48, 4, 30, 78, 49, 19, 78, 38, 86, 39, 43, 18, 5, 55, 21, 49, 10, 49, 85, 43, 78}

Out[103]= {37, 38, 37, 38, 38, 38, 39}

In[104]:=
```
RandomInteger[100, 100]
Cases[%, x_ /; (40 > x > 36 && OddQ[x])]
```
losowa liczba całkowita / przypadki / liczba nieparz

Out[104]= {25, 45, 30, 57, 35, 37, 64, 33, 50, 98, 63, 88, 70, 44, 47, 86, 36, 40, 93, 33, 58, 85, 9, 4, 31, 18, 33, 96, 2, 17, 77, 12, 80, 2, 75, 19, 82, 87, 16, 8, 0, 72, 47, 31, 64, 26, 12, 80, 21, 13, 74, 45, 26, 74, 59, 38, 60, 62, 36, 20, 50, 83, 65, 63, 77, 95, 7, 28, 28, 27, 60, 51, 91, 16, 76, 83, 65, 85, 55, 62, 65, 36, 48, 21, 28, 13, 78, 92, 36, 29, 59, 100, 24, 54, 94, 34, 64, 84, 20, 10}

Out[105]= {37}

Wielomiany Legendre’a

W definicji funkcji Legendre’a |x|<1

In[106]:=
```
Clear[P]
P[n_Integer?NonNegative, x_Real /; x^2 <= 1] :=
  (1/(2^n n!)) D[(x^2 - 1)^n, {x, n}] // Simplify
P[n_Integer?NonNegative, x_Symbol] :=
  (1/(2^n n!)) D[(x^2 - 1)^n, {x, n}] // Simplify
Pbis[n_Integer?NonNegative, x_Real] :=
  (1/(2^n n!)) Derivative[n][(#^2 - 1)^n &][x] // Simplify
Pbis[n_Integer?NonNegative, x_Symbol] :=
  (1/(2^n n!)) Derivative[n][(#^2 - 1)^n &][x] // Simplify
```
wyczyść / wprowadź ··· / nieujemny / część rzeczywista / oblicz pochodną / uprość / pochodna

In[111]:=
```
P[2, x]
P[2, 0.99]
Pbis[2, x]
Pbis[2, 0.99]
```

Out[111]= $\frac{1}{2}(3x^2-1)$

General: 0.99 is not a valid variable.

Out[112]= $\frac{1}{8}\,\frac{\partial^2 0.00039601}{\partial 0.99^2}$

Out[113]= $\frac{1}{2}(3x^2-1)$

Out[114]= 0.97015

Widzimy moc funkcji Derivative!

In[115]:=
```
Clear[P]
P[n_Integer?NonNegative, x_Real /; x^2 <= 1] :=
  (1/(2^n n!)) D[(y^2 - 1)^n, {y, n}] /. y → x // Simplify
P[n_Integer?NonNegative, x_Symbol] :=
  (1/(2^n n!)) D[(x^2 - 1)^n, {x, n}] // Simplify
```

In[118]:=
```
P[2, 0.99]
```

Out[118]= 0.97015

In[119]:=
```
LegendreP[1.5, 3]
```

Out[119]= 6.11134

In[120]:=
```
Pnew[n_Integer?NonNegative, x_Real] := LegendreP[n, x]
Pnew[n_Integer?NonNegative, x_Symbol] := LegendreP[n, x]
```

In[122]:=
```
Pnew[2, 0.5]
```

Out[122]= -0.125

In[123]:=
```
Pnew[3, x]
```

Out[123]= $\frac{1}{2}\left(5 x^3 - 3 x\right)$

In[124]:=
```
Clear[P, Pnew]
```

Warunki w definicjach reguł: := oraz :>

In[125]:=
```
lhs :→ /; warunek
RuleDelayed
lhs := /; warunek
(*SetDelayed *)
```

Operację są wykonywane jeśli warunek jest spełniony

:→ :>

In[125]:=
```
{f[2], f[a], g[6], f[5]} /. f[x_] :> x /; NumberQ[x]
```
liczba?

Out[125]= $\{2, f(a), g(6), 5\}$

In[126]:=
```
{f[2], f[a], g[6], f[5]} /. _[x_] :> x /; NumberQ[x]
```
liczba?

Out[126]= $\{2, f(a), 6, 5\}$

In[127]:=
```
(1 + x) ^3 - (1 + x) ^2 // Expand
% /. x_Integer :> A /; OddQ[x]
%% /. x_Integer → A /; OddQ[x]
```
rozszerz; wprowadź komende; liczba niepar:; wprowadź komende; liczba niep

Out[127]= $x^3 + 2x^2 + x$

Out[128]= $x^A + 2x^2 + x$

Out[129]= $(A\ /;\ \mathrm{OddQ}[2])\, x^{A/;\mathrm{OddQ}[2]} + x^{A/;\mathrm{OddQ}[3]} + x$

In[130]:=
```
:>
```

In[130]:=
```
:> vs. →
```

In[130]:=
```
{f[2], f[a], g[6], f[5]} /. f[x_] → x /; NumberQ[x]
```
liczba?

Out[130]= $\{2\ /;\ \mathrm{NumberQ}[2], a\ /;\ \mathrm{NumberQ}[a], g(6), 5\ /;\ \mathrm{NumberQ}[5]\}$

In[131]:=
```
Clear[f]
f[x_, y_] := Sqrt[x - y] /; x > y
```
wyczyść; pierwiastek kwadratowy

In[133]:=
```
f[5, 3]
f[3, 5]
```

Out[133]= $\sqrt{2}$

Out[134]= $f(3, 5)$

In[135]:=
```
Clear[f2]
f2[x_, y_] = Sqrt[x - y] /; x > y
```
wyczyść
pierwiastek kwadratowy

Out[136]= $\sqrt{x-y}$ /; $x > y$

In[137]:=
```
f2[5, 3]
f2[3, 5]
```

Out[137]= $\sqrt{2}$ /; $5 > 3$

Out[138]= $i\sqrt{2}$ /; $3 > 5$

Równoważność struktur

In[139]:=
```
? MatchQ
```

MatchQ[*expr*, *form*] returns True if the pattern *form* matches *expr*, and returns False otherwise.
MatchQ[*form*] represents an operator form of MatchQ that can be applied to an expression. >>

In[140]:=
```
MatchQ[1, 2]
```
pasuje do wzoru?

Out[140]= False

In[141]:=
```
MatchQ[a, b]
```
pasuje do wzoru?

Out[141]= False

In[142]:=
```
MatchQ[a, a]
```
pasuje do wzoru?

Out[142]= True

In[143]:=
```
MatchQ[a^2 - b^2, x^2 - y^2]
```
pasuje do wzoru?

Out[143]= False

In[144]:=
```
MatchQ[a^2 - b^2, x_^2 - y_^2]
```
pasuje do wzoru?

Out[144]= True

In[145]:=
```
MatchQ[Sin[a]^2 - Sin[b]^2, x_^2 - y_^2]
pasuje⋯ sinus      sinus
```
Out[145]= True

In[146]:=
```
MatchQ[(a - b) (a + b), x_^2 - y_^2]
pasuje do wzoru?
```
Out[146]= False

In[147]:=
```
MatchQ[a - b, x_ - y_]
pasuje do wzoru?
```
Out[147]= True

In[148]:=
```
MatchQ[a - 2 b, x_ - y_]
pasuje do wzoru?
```
Out[148]= False

Wartość domyślna

In[149]:=
```
x_: X (* jeśli nie nic nie wstawimy w miejsce x to wstawione zostanie X! *)
```
Out[149]= x_ : X

In[150]:=
```
{a, b} /. {x_, y_} → {x^2, y^2}
```
Out[150]= $\{a^2, b^2\}$

In[151]:=
```
{a, b} /. {x_, y_} :> {x^2, y^2}
```
Out[151]= $\{a^2, b^2\}$

In[152]:=
```
{a} /. {x_, y_} :> {x^2, y^2}
```
Out[152]= $\{a\}$

In[153]:=
```
{a} /. {x_, y_: d} :> {x^2, y^2}
```
Out[153]= $\{a^2, d^2\}$

In[154]:=
```
{a, 2} /. {x_, y_Integer: 10} :> {x^2, y^2}
```
Out[154]= $\{a^2, 4\}$

In[155]:=
```
{a} /. {x_, y_Integer: 10} :> {x^2, y^2}
```
Out[155]= $\{a^2, 100\}$

In[156]:=
```
Clear[F]
wyczyść
F[x_: 9] := x^2
```

In[158]:=
```
F[]
```
Out[158]= 81

In[159]:=
```
Clear[F]
wyczyść
F[x_: 0, y_] := x^2 + y
```

In[161]:=
```
F[y]
```
Out[161]= y

In[162]:=
```
F[]
```
Out[162]= $F()$

x_. wbudowana wartość domyślna

In[163]:=
```
x_. (* wbudowana wartść domyślna  *)
```
Out[163]= x_.

In[164]:=
```
x_ + y_. (* wbudowana wartść wynosi 0 *)
```
Out[164]= x_ + y_.

In[165]:=
```
a /. x_ + y_. :> x G[y]
```
Out[165]= $a\,G(0)$

In[166]:=
```
a /. x_ y_. :> x G[y]
```
Out[166]= $a\,G(1)$

Wesołych Świąt

In[230]:=

```
PD = .5;
s[t_, f_] := t^.6 - f
dt[cl_, ps_, sg_, hf_, dp_, f_, flag_] :=
 Module[{sv, basePt}, {PointSize[ps], sv = s[t, f];
 moduł                 rozmiar kropki
   Hue[cl (1 + Sin[.02 t]) / 2, 1, .3 + sg .3 Sin[hf sv]],
   odcień      sinus                        sinus
   basePt = {-sg s[t, f] Sin[sv], -sg s[t, f] Cos[sv], dp + sv};
                            sinus            cosinus
   Point[basePt], If[flag, {Hue[cl (1 + Sin[.1 t]) / 2, 1, .6 + sg .4 Sin[hf sv]],
   punkt          operator w··· odcień  sinus                      sinus
     PointSize[RandomReal[.01]], Point[basePt + 1 / 2 RotationTransform[
                losowa liczba rzeczywi··· punkt      przekształcenie obrotowe
          20 sv, {-Cos[sv], Sin[sv], 0}][{Sin[sv], Cos[sv], 0}]]}, {}]}]
                  cosinus   sinus          sinus    cosinus

frames = ParallelTable[
           tabela równoległa
   Graphics3D[Table[{dt[1, .01, -1, 1, 0, f, True], dt[.45, .01, 1, 1, 0, f, True],
   trójwymiarow·· tabela                       prawda                          prawda
      dt[1, .005, -1, 4, .2, f, False], dt[.45, .005, 1, 4, .2, f, False]},
                                 fałsz                            fałsz
     {t, 0, 200, PD}], ViewPoint → Left, BoxRatios → {1, 1, 1.3}],
                       punkt widzenia lewy proporcje obudowy
    ViewVertical → {0, 0, -1}, ViewCenter → {{0.5, 0.5, 0.5}, {0.5, 0.55}},
    ustawienie w pionie        środek widzenia
    Boxed → False, PlotRange → {{-20, 20}, {-20, 20}, {0, 20}},
    dodaj o··· fałsz zakres wykresu
    Background → Black], {f, 0, 1, .01}];
    tło          czarny
ListAnimate[%]
animuj liste
(* Export["tree.gif",frames]*)
   eksportuj
```

Out[234]=