Functii si proceduri

1. Se considera $x^3 - (a-1)\,x^2 + a^2\,x - a^3 = 0$ ca o ecuatie in x. Rezolvati-o, faceti o functie din prima solutie si calculati solutia pentru $a=0$ si $a=1$. Dati o solutie aproximativa pentru $a=2$.

```
> restart;
> x^3-(a-1)*x^2+a^2*x-a^3=0;
```

$$x^3 - (a-1)\,x^2 + a^2\,x - a^3 = 0 \qquad \textbf{(1.1)}$$

```
> sols := solve(%, x);
```

$sols :=$

$$\frac{1}{6}\Big(80\,a^3 + 12\,a^2 + 24\,a - 8 + 12\sqrt{48\,a^6 + 24\,a^5 - 12\,a^3 + 33\,a^4}\Big)^{1/3} - \Big(6\Big(\frac{2}{9}\,a^2 + \frac{2}{9}\,a - \frac{1}{9}\Big)\Big) \Big/ \Big(80\,a^3 + 12\,a^2 + 24\,a - 8 + 12\sqrt{48\,a^6 + 24\,a^5 -$$

$$/3 + \frac{a}{3} - \frac{1}{3},$$

$$-\frac{1}{12}\Big(80\,a^3 + 12\,a^2 + 24\,a - 8 + 12\sqrt{48\,a^6 + 24\,a^5 - 12\,a^3 + 33\,a^4}\Big)^{1/3} + \Big(3\Big(\frac{2}{9}\,a^2 + \frac{2}{9}\,a - \frac{1}{9}\Big)\Big) \Big/ \Big(80\,a^3 + 12\,a^2 + 24\,a - 8 + 12\sqrt{48\,a^6 + 24\,a^5 -$$

$$/3 + \frac{a}{3} - \frac{1}{3}$$

$$+ \frac{1}{2}\,\mathrm{I}\sqrt{3}\,\Bigg(\frac{1}{6}\Big(80\,a^3 + 12\,a^2 + 24\,a - 8 + 12\sqrt{48\,a^6 + 24\,a^5 - 12\,a^3 + 33\,a^4}\Big)^{1/3}$$

$$+ \Big(6\Big(\frac{2}{9}\,a^2 + \frac{2}{9}\,a - \frac{1}{9}\Big)\Big) \Big/ \Big(80\,a^3 + 12\,a^2 + 24\,a - 8 + 12\sqrt{48\,a^6 + 24\,a^5 -$$

$$/3\Bigg),$$

$$-\frac{1}{12}\Big(80\,a^3+12\,a^2+24\,a-8$$
$$+12\sqrt{48\,a^6+24\,a^5-12\,a^3+33\,a^4}\Big)^{1/3}$$
$$+\left(3\left(\frac{2}{9}\,a^2+\frac{2}{9}\,a-\frac{1}{9}\right)\right)\Big/\Big(80\,a^3+12\,a^2+24\,a-8+12\sqrt{48\,a^6+24\,a^5-}$$
$$/3$$
$$+\frac{a}{3}-\frac{1}{3}$$
$$-\frac{1}{2}\,\mathrm{I}\sqrt{3}\left(\frac{1}{6}\Big(80\,a^3+12\,a^2+24\,a-8\right.$$
$$+12\sqrt{48\,a^6+24\,a^5-12\,a^3+33\,a^4}\Big)^{1/3}$$
$$+\left(6\left(\frac{2}{9}\,a^2+\frac{2}{9}\,a-\frac{1}{9}\right)\right)\Big/\Big(80\,a^3+12\,a^2+24\,a-8+12\sqrt{48\,a^6+24\,a^5-}$$
$$\left./3\right)$$

```
> f := unapply(%[1], a);
```

$$f:=a\rightarrow\frac{1}{6}\Big(80\,a^3+12\,a^2+24\,a-8$$
$$+12\sqrt{48\,a^6+24\,a^5-12\,a^3+33\,a^4}\Big)^{1/3}$$
$$-\left(6\left(\frac{2}{9}\,a^2+\frac{2}{9}\,a-\frac{1}{9}\right)\right)\Big/\Big(80\,a^3+12\,a^2+24\,a-8+12\sqrt{48\,a^6+24\,a^5-}$$
$$/3$$
$$+\frac{1}{3}\,a-\frac{1}{3}$$

```
> f(0);
```

$$\frac{(-8)^{1/3}}{6}-\frac{(-8)^{2/3}}{12}-\frac{1}{3} \qquad \textbf{(1.4)}$$

```
> simplify(%);
```

$$0 \qquad \textbf{(1.5)}$$

```
> f(1);
```

$$\frac{\left(108+12\sqrt{93}\right)^{1/3}}{6}-\frac{2}{\left(108+12\sqrt{93}\right)^{1/3}} \qquad \textbf{(1.6)}$$

```
> f(2.0);
```

$$1.607521768 \qquad \textbf{(1.7)}$$

2. Definiti o functie in Maple care ia valoarea 1 pe intervalul [-1,1] si 0 in rest.

Reprezentati-o grafic.

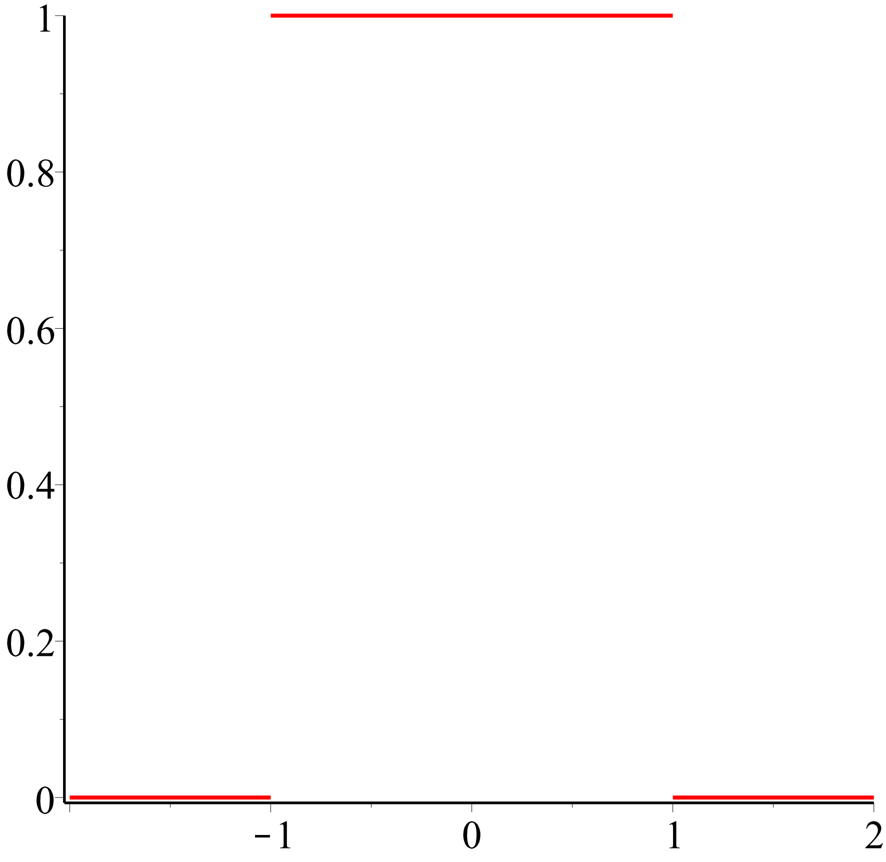
```
> restart;
> f := x -> piecewise(x<-1, 0, x<=1, 1, 0);
```

$$f := x \rightarrow piecewise(x < -1, 0, x \leq 1, 1, 0) \tag{2.1}$$

```
> f(x);
```

$$\begin{cases} 0 & x < -1 \\ 1 & x \leq 1 \\ 0 & otherwise \end{cases} \tag{2.2}$$

```
> plot(f, -2..2, axes=frame,discont=true);
```



Puteti defini functia dumneavoastra cu ajutorul functiei lui Heaviside:

```
> convert(f(x), Heaviside);
```

$$\mathrm{Heaviside}(1 + x) - \mathrm{Heaviside}(-1 + x) \tag{2.3}$$

Se poate si mai complicat, de exemplu

```
> g := x -> max(-x^2+1,0)/(1-x^2);
```

(2.4)

$$g := x \to \frac{\max(-x^2+1, 0)}{-x^2+1} \qquad \textbf{(2.4)}$$

```
> g(x);
```

$$\frac{\max(0, -x^2+1)}{-x^2+1} \qquad \textbf{(2.5)}$$

```
> convert(%, piecewise);
```

$$\begin{cases} 0 & x \le -1 \\ 1 & x < 1 \\ 0 & 1 \le x \end{cases} \qquad \textbf{(2.6)}$$

3. Definiti functia $f: t \to \sum_{n=1}^{8} (-1)^{n+1} \frac{2}{n} \sin(n\, t)$. Calculati $f\left(\frac{\pi}{10}\right)$ si $f\left(\frac{\pi}{6}\right)$. Desenati graficul functiei.

```
> restart:
> f := t-> sum((-1)^(n+1)*2/n*sin(n*t), n=1..8);
```

$$f := t \to \sum_{n=1}^{8} \frac{2\,(-1)^{n+1} \sin(n\,t)}{n} \qquad \textbf{(3.1)}$$

```
> f(Pi/10);
```

$$\frac{2}{5} + 2\sin\left(\frac{\pi}{10}\right) - \frac{5}{4}\sin\left(\frac{\pi}{5}\right) + \frac{20}{21}\sin\left(\frac{3\pi}{10}\right) - \frac{5}{6}\sin\left(\frac{2\pi}{5}\right) \qquad \textbf{(3.2)}$$

```
> convert(%, radical);
```

$$\frac{29}{210} + \frac{31\sqrt{5}}{42} - \frac{5\sqrt{2}\sqrt{5-\sqrt{5}}}{16} - \frac{5\sqrt{2}\sqrt{5+\sqrt{5}}}{24} \qquad \textbf{(3.3)}$$

```
> evalf(%);
```

$$0.2612477025 \qquad \textbf{(3.4)}$$

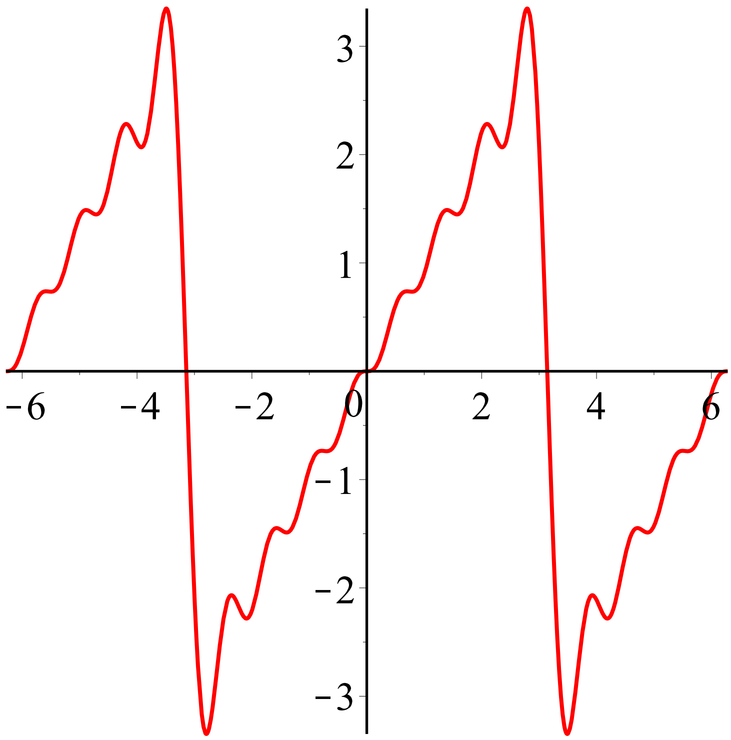
```
> f(Pi/6);
```

$$\frac{181}{105} - \frac{5\sqrt{3}}{8} \qquad \textbf{(3.5)}$$

```
> evalf(%);
```

$$0.641277769 \qquad \textbf{(3.6)}$$

```
> plot(f, -2*Pi..2*Pi);
```



4. Cat va fi f(1), f(4), si f() dupa atribuirile urmatoare

```
> x := 1:
> f := proc(x) 2 end proc:
> f(x) := 3: ?
```

```
> restart;
> x := 1:
> f := proc(x) 2 end proc:
> f(x) := 3:
> f(1);
```

$$3 \tag{4.1}$$

```
> f(4);
```

$$2 \tag{4.2}$$

```
> f();
```

$$2 \tag{4.3}$$

f este egala cu constanta 2 cu exceptia punctului 1 unde f(1) este egal cu 3.

5. Scrieti o procedura Maple ce calculeaza polinoamele lui Legendre $L_n(x)$. Aceste polinoame satisfac relatia de recurenta $L_0(x) = 1$, $L_1(x) = x$, si

$L_n(x) = \frac{n-1}{n} \left(x\,L_{n-1}(x) - L_{n-2}(x)\right) + x\,L_{n-1}(x)$, pentru n `>` 1.

Calculati $L_7(x)$ si vericati raspunsul cu procedura Maple **LegendreP**. Poate procedura dumneavoastra sa calculeze $L_{50}(x)$?

```
> restart;
> L := proc(n::nonnegint, x::anything) Legendre(n,
  x) end:
> Legendre := proc(n,x)
    option remember;
    if n=0 then
      1
    elif n=1 then
      x
    else
      expand( (n-1)/n*(x*L(n-1,x) - L(n-2,x)) + x*L
  (n-1,x) )
    end if
  end proc;
```

```
Legendre := proc(n, x)
   option remember;
   if n = 0 then
      1
   elif n = 1 then
      x
   else
      expand((n - 1) * (x * L(n - 1, x) - L(n - 2, x)) / n + x
      * L(n - 1, x))
   end if
end proc
```
(6.1)

```
> L(7,x);
```

$$\frac{429}{16}x^7 - \frac{693}{16}x^5 + \frac{315}{16}x^3 - \frac{35}{16}x \quad \textbf{(6.2)}$$

```
> simplify(LegendreP(7,x));
```

$$\frac{429}{16}x^7 - \frac{693}{16}x^5 + \frac{315}{16}x^3 - \frac{35}{16}x \quad \textbf{(6.3)}$$

```
> L(50,x);
```

(6.4)

$$
\begin{aligned}
&\frac{20146690401016725}{140737488355328}\,x^{2}-\frac{1067774591253886425}{35184372088832}\,x^{4} \qquad \textbf{(6.4)}\\
&\quad+\frac{9004899052907775175}{35184372088832}\,x^{6}-\frac{15801325804719}{140737488355328}\\
&\quad-\frac{8065816723104536070675}{70368744177664}\,x^{8}\\
&\quad+\frac{222078820442811559812585}{70368744177664}\,x^{10}\\
&\quad-\frac{2052546673789621992207225}{35184372088832}\,x^{12}\\
&\quad+\frac{26998883170617335435956575}{35184372088832}\,x^{14}\\
&\quad-\frac{1052956443654076082002306425}{140737488355328}\,x^{16}\\
&\quad+\frac{7838675747202566388239392275}{140737488355328}\,x^{18}\\
&\quad-\frac{5693353963757653481984400705}{17592186044416}\,x^{20}\\
&\quad+\frac{26248579962778792027330678575}{17592186044416}\,x^{22}\\
&\quad-\frac{194391657405506706173419952925}{35184372088832}\,x^{24}\\
&\quad+\frac{583174972216520118520259858775}{35184372088832}\,x^{26}\\
&\quad-\frac{712769410486857922635873160725}{17592186044416}\,x^{28}\\
&\quad+\frac{1423900270604780539702468452115}{17592186044416}\,x^{30}\\
&\quad-\frac{18602568051449552212241926551825}{140737488355328}\,x^{32}\\
&\quad+\frac{24770264410753681822717859419275}{140737488355328}\,x^{34}\\
&\quad-\frac{6684039602901787158511168414725}{35184372088832}\,x^{36}\\
&\quad+\frac{5790298887862287879848224131675}{35184372088832}\,x^{38}\\
&\quad-\frac{7928255400303748020099876118755}{70368744177664}\,x^{40}\\
&\quad+\frac{4189728463575151392735706892025}{70368744177664}\,x^{42}\\
&\quad-\frac{823773249709279237895181270525}{35184372088832}\,x^{44}\\
&\quad+\frac{226836112238787036521861509275}{35184372088832}\,x^{46}
\end{aligned}
$$

$$-\frac{156050375086257748529223875175}{140737488355328}x^{48}+\frac{12611418068195524166851562157}{140737488355328}x^{50}$$

```
> sort(%);
```

$$\begin{aligned}
&\frac{12611418068195524166851562157}{140737488355328}x^{50}-\frac{156050375086257748529223875175}{140737488355328}x^{48}+\frac{226836112238787036521861509275}{35184372088832}x^{46}\\
&-\frac{823773249709279237895181270525}{35184372088832}x^{44}+\frac{4189728463575151392735706892025}{70368744177664}x^{42}\\
&-\frac{7928255400303748020099876118755}{70368744177664}x^{40}+\frac{5790298887862287879848224131675}{35184372088832}x^{38}\\
&-\frac{6684039602901787158511168414725}{35184372088832}x^{36}+\frac{24770264410753681822717859419275}{140737488355328}x^{34}\\
&-\frac{18602568051449552212241926551825}{140737488355328}x^{32}+\frac{1423900270604780539702468452115}{17592186044416}x^{30}\\
&-\frac{712769410486857922635873160725}{17592186044416}x^{28}+\frac{583174972216520118520259858775}{35184372088832}x^{26}\\
&-\frac{194391657405506706173419952925}{35184372088832}x^{24}+\frac{26248579962778792027330678575}{17592186044416}x^{22}\\
&-\frac{5693353963757653481984400705}{17592186044416}x^{20}+\frac{7838675747202566388239392275}{140737488355328}x^{18}\\
&-\frac{1052956443654076082002306425}{140737488355328}x^{16}+\frac{26998883170617335435956575}{35184372088832}x^{14}
\end{aligned} \tag{6.5}$$

$$-\frac{20525466737896219922207225}{35184372088832}x^{12}+\frac{22207882044281155981 2585}{70368744177664}x^{10}-\frac{80658167231045360706 75}{70368744177664}x^{8}+\frac{90048990529077755175}{35184372088832}x^{6}-\frac{1067774591253886425}{35184372088832}x^{4}+\frac{20146690401016725}{140737488355328}x^{2}-\frac{15801325804719}{140737488355328}$$

6. Scrieti o functie anonima care selecteaza dintr-o multime de intregi elementele care au valori intre 0 si 10. Utilizati procedura **rand** pentru a genera 100 de intregi aleatori si aplicati functia dumneavoastra anonima acestei multimi.

```
> restart;
> die := rand(-50..50):
> {seq(die(), k=1..100)};
```

$\{-50, -49, -48, -47, -44, -43, -42, -41, -40, -39, -36, -35, -34, -30, -29, -28, -27, -26, -24, -23, -21, -20, -18, -17, -15, -14, -13, -11, -9, -8, -4, -3, 0, 1, 3, 5, 6, 8, 10, 11, 12, 13, 15, 16, 17, 20, 21, 23, 24, 25, 26, 28, 32, 33, 34, 35, 37, 39, 40, 42, 44, 46, 47, 49, 50\}$ **(7.1)**

Functia anonima ceruta ar putea fi $n \to 0 < n$ **and** $n < 10$.

```
> select(n -> n>0 and n<10, %);
```

$\{1, 3, 5, 6, 8\}$ **(7.2)**

Fara o functie anonima am fi putut scrie

```
> select(has, %%, {seq(k,k=1..9)});
```

$\{1, 3, 5, 6, 8\}$ **(7.3)**

7. Scrieti o functie anonima care elimina termenii cu coeficienti negativi dintr-un polinom de doua variabile. (Polinomul se poate crea cu procedura **randpoly**).

```
> restart;
> randpoly([x,y], coeffs=rand(-4..4));
```

$x - 4y^2 + 3x^3y - 3y^4$ **(8.1)**

Functia anonima ceruta ar putea fi $t \to lcoeff(t)$ \`>\`$\cdot 0$.

```
> select(t->lcoeff(t)>0, %);
```

$x + 3x^3y$ **(8.2)**

8. Definiti o functie `midpoint`, care returneaza media a doua argumente date la intrare. De exemplu, `midpoint(2,3)` returneaza 5/2, `midpoint(a,b)` returneaza `a/2 + b/2`.

Solutie

```
> restart;
> midpoint:=(x,y)->(x+y)/2;
```

$$midpoint := (x, y) \rightarrow \frac{1}{2}\, x + \frac{1}{2}\, y \tag{9.1.1}$$

```
> midpoint(2,3);
```

$$\frac{5}{2} \tag{9.1.2}$$

```
> midpoint(a,b);
```

$$\frac{a}{2} + \frac{b}{2} \tag{9.1.3}$$

9. Pentru doua numere naturale a si b definiti cel mai mic multiplu comun al lor $lcm(a, b)$ cu ajutorul celui mai mare divizor comun $gcd(a, b)$ astfel: $lcm(a, b) = \frac{a\, b}{gcd(a, b)}$. Daca unul dintre numere sau amandoua sunt zero, definim $lcm(a, b) = 0$. Utilizati operatorul sageata (arrow operator) si piecewise pentru a defini functia `my_lcm` care returneaza cel mai mic multiplu comun a doua numere naturale. Pentru test, alegeti numere naturale aleatoare si comparati cu rezultatul furnizat de procedura `ilcm`. Testati si cazul cand argumentele de intrare sunt zero.

Solutie

```
> restart;
> my_lcm:=(a,b)->piecewise(a=0,0,b=0,0,a*b/igcd
  (a,b));
```

$$my\_lcm := (a, b) \rightarrow piecewise\left(a = 0, 0, b = 0, 0, \frac{a\, b}{igcd(a, b)}\right) \tag{10.1.1}$$

```
> ga:=rand(0..100); gb:=rand(0..100);
```

$ga :=$ **proc**( ) **(10.1.2)**

proc( )

option $builtin = RandNumberInterface$;

end proc(6, 101, 7)

end proc
$gb := \mathbf{proc}(\ )$
    proc( )
        option builtin = RandNumberInterface;
    end proc(6, 101, 7)
end proc

```
> a:=ga(); b:=gb();
```

$$a := 92$$
$$b := 44 \qquad (10.1.3)$$

```
> my_lcm(a,b); ilcm(a,b);
```

$$1012$$
$$1012 \qquad (10.1.4)$$

```
> my_lcm(0,b); ilcm(0,b); my_lcm(a,0); my_lcm(0,
  0); ilcm(0,0);
```

$$0 \qquad (10.1.5)$$
$$0$$
$$0$$
$$0$$
$$0$$

10. Se considera polinomul de gradul al doilea $p = a\,x^2 + b\,x + c$ cu coeficientii ca parametri. Presupunem ca dorim sa cream o functie generala de x, care calculeaza valoarea polinomului in x. De exemplu, *f(1)* returneaza $a + b + c$, *f(2)* returneaza $4a + 2b + c$, s.a.m.d. Introduceti `p := a*x^2 + b*x + c` si, *fara a retipari formula* pentru `p`, creati *f* **in patru moduri diferite**. De fiecare data testati functia dumneavoastra pentru cateva valori ale lui *x*.

Solutie

```
> restart;
> p:=a*x^2+b*x+c;
```

$$p := a\,x^2 + b\,x + c \qquad (11.1.1)$$

Cu unapply

```
> f1:=unapply(p,x);
```

$$f1 := x \rightarrow a\,x^2 + b\,x + c \qquad (11.1.2)$$

```
> f1(1), f1(2);
```

$$a + b + c, 4\,a + 2\,b + c \qquad (11.1.3)$$

Cu codegen[makeproc]

```
> with(codegen);
```

$$[C, GRAD, GRADIENT, HESSIAN, JACOBIAN, MathML, cost, declare, dontreturn, eqn, fortran, horner, intrep2maple, joinprocs, makeglobal, makeparam, makeproc, makevoid, \qquad (11.1.4)$$

maple2intrep, optimize, packargs, packlocals, packparams, prep2trans, renamevar, split, swapargs]

```
> f2:=makeproc(p,x);
```

$$f2 := \mathbf{proc}(x)\ \ a * x\hat{}2 + b * x + c\ \ \mathbf{end\ proc} \tag{11.1.5}$$

```
> f2(1),f2(2);
```

$$a + b + c, 4\,a + 2\,b + c \tag{11.1.6}$$

Cu procedura si evaluare

```
> f3:=proc(v) global p; eval(p,x=v); end;
```

$$f3 := \mathbf{proc}(v)\ \ \mathbf{global}\ p;\ eval(p, x = v)\ \ \mathbf{end\ proc} \tag{11.1.7}$$

```
> f3(1),f3(2),f3(3);
```

$$a + b + c, 4\,a + 2\,b + c, 9\,a + 3\,b + c \tag{11.1.8}$$

Cu procedura si substitutie

```
> f4:=proc(v) global p; eval(subs(x=v,p)); end
  proc;
```

$$f4 := \mathbf{proc}(v)\ \ \mathbf{global}\ p;\ eval(subs(x = v, p))\ \ \mathbf{end\ proc} \tag{11.1.9}$$

```
> f4(1),f4(2),f4(3);
```

$$a + b + c, 4\,a + 2\,b + c, 9\,a + 3\,b + c \tag{11.1.10}$$

11. Se considera formula de rezolvare a ecuatiei de gradul 2 `p := a*x^2 + b*x + c=0`.
(a) Utilizati iesirea comenzii `solve(p,x);` pentru a crea o functie care returneaza prima radacina, in functie de parametrii a, b, and c.
(b) Din iesirea lui `solve` selectati discriminantul $b^2 - 4\,a\,c$ (utilizati comanda `op`) si creati o functie care returneaza valoarea discriminantului pentru a, b si c dati.
(c) Formula pentru prima radacina este valida doar pentru $a \neq 0$. Utilizati `piecewise` pentru a crea o functie care testeaza cazul $a = 0$ si care returneaza in acest caz $-\frac{c}{b}$ si in caz contrar apeleaza functia de la punctul (a).
(d) Extindeti functia creata la punctul (c) pentru a trata cazul $b = 0$. In cazul $b = 0$, functia va returna ∞, sau simbolul `infinity`, altfel se va apela functia creata la punctul (c).

(a)

```
> restart;
```

```
> p:=a*x^2+b*x+c;
```

$$p := a\,x^2 + b\,x + c \tag{12.1.1}$$

```
> s:=solve(p,x);
```

$$s := -\frac{b - \sqrt{b^2 - 4ac}}{2a},\ -\frac{b + \sqrt{b^2 - 4ac}}{2a} \qquad (12.1.2)$$

```
> f:=unapply(s[1],a,b,c);
```

$$f := (a, b, c) \rightarrow -\frac{1}{2}\,\frac{b - \sqrt{b^2 - 4ac}}{a} \qquad (12.1.3)$$

```
> f(1,1,1);
```

$$-\frac{1}{2} + \frac{\sqrt{-3}}{2} \qquad (12.1.4)$$

(b)

```
> op(s[1]);
```

$$-\frac{1}{2},\ b - \sqrt{b^2 - 4ac},\ \frac{1}{a} \qquad (12.2.1)$$

```
> eee:=op([2,2,2,1],s[1]);
```

$$eee := b^2 - 4ac \qquad (12.2.2)$$

```
> discr:=unapply(eee,a,b,c);
```

$$discr := (a, b, c) \rightarrow b^2 - 4ac \qquad (12.2.3)$$

```
> discr(1,1,2);
```

$$-7 \qquad (12.2.4)$$

(c)

```
> f2:=proc(a,b,c) piecewise(a=0, -c/b, f(a,b,c));
  end;
```

$f2 :=$ **proc**(a, b, c) $piecewise(a = 0,\ -c/b, f(a, b, c))$ **end proc** (12.3.1)

```
> f2(1,1,1);
```

$$-\frac{1}{2} + \frac{\sqrt{-3}}{2} \qquad (12.3.2)$$

```
> f2(0,1,-2);
```

$$2 \qquad (12.3.3)$$

(d)

```
> f3:=proc(a,b,c) piecewise(a=0 and b=0,
  infinity, f2(a,b,c)); end proc;
```

$f3 :=$ **proc**(a, b, c) (12.4.1)

$piecewise(a = 0$ **and** $b = 0, infinity, f2(a, b, c))$

```
end proc
> f3(1,1,1);
```

$$-\frac{1}{2}+\frac{\sqrt{-3}}{2} \qquad (12.4.2)$$

```
> f3(0,1,-2);
```

$$2 \qquad (12.4.3)$$

```
> f3(1,0,1);
```

$$\frac{\sqrt{-4}}{2} \qquad (12.4.4)$$

```
> simplify(%);
```

$$I \qquad (12.4.5)$$

```
> f3(0,0,2);
```

$$\infty \qquad (12.4.6)$$

12. Functia exponentiala e^x poate fi aproximata prin

$$\sum_{k=0}^{N} \frac{x^k}{k!},\ N \text{ numar natural.}$$

Creati o functie **`expfun`** cu doua argumente, N si x. Cat de mare trebuie sa fie N la apelul **`expfun(N,-0.1)`** pentru a coincide cu 10 cifre zecimale cu iesirea lui **`exp(-0.1)`**.

Solutie

```
> restart; Digits:=12;
```

$$Digits := 12 \qquad (13.1.1)$$

```
> expfun:=(N,x)->add(x^k/k!,k=0..N);
```

$$expfun := (N, x) \rightarrow add\left(\frac{x^k}{k!}, k=0..N\right) \qquad (13.1.2)$$

```
> expfun(10,-0.1);
```

$$0.904837418036 \qquad (13.1.3)$$

```
> exp(-0.1);
```

$$0.904837418036 \qquad (13.1.4)$$

```
> expfun(9,-0.1);
```

$$0.904837418036 \qquad (13.1.5)$$

```
> expfun(8,-0.1);
```

$$0.904837418036 \qquad (13.1.6)$$

```
> expfun(7,-0.1);
```

$$0.904837418036 \qquad (13.1.7)$$

```
> expfun(6,-0.1);
```

$$0.904837418056 \quad (13.1.8)$$

```
> eval((0.1)^(N+1)/(N+1)!,N=6); eval((0.1)^(N+1)/
  (N+1)!,N=5);
```

$$1.98412698413\ 10^{-11} \quad (13.1.9)$$

$$1.38888888889\ 10^{-9}$$

13. Inaltimea unui polinom cu coeficienti intregi se defineste ca fiind cel mai mare coeficient. De exemplu,

```
> p := randpoly(x);
> coeffs(p);
> max(%);
```

determina inaltimea unui polinom aleator. Utilizati comenzile **coeffs** si **max** pentru a defini o functie **height** care returneaza inaltimea unui polinom. Testati daca **height** functioneaza pentru polinoame in mai multe variabile.

Solutie

```
> height:=proc(p::polynom)
  max(coeffs(p));
  end proc;
```

$$height := \mathbf{proc}(p::polynom)\ \max(coeffs(p))\ \mathbf{end\ proc} \quad (14.1.1)$$

```
> p:=randpoly(x);;
```

$$p := 97 - 94\,x^5 + 87\,x^4 - 56\,x^3 - 62\,x \quad (14.1.2)$$

```
> height(p);
```

$$97 \quad (14.1.3)$$

```
> q:=randpoly([x,y]);
```

$$q := -44\,x + 71\,x\,y - 17\,x^2\,y - 75\,x^4\,y - 10\,x^3\,y^2 - 7\,x\,y^4 \quad (14.1.4)$$

```
> coeffs(q);
```

$$-44, 71, -17, -75, -10, -7 \quad (14.1.5)$$

```
> height(q);
```

$$71 \quad (14.1.6)$$

```
> height(sqrt(t));
Error, invalid input: height expects its 1st argument,
p, to be of type polynom, but received t^(1/2)
```

14. Scrieti o procedura **fractional_power** care returneaza $x^{\frac{1}{n}}$ pentru un argument x si un index n. Daca indexul lipseste, **fractional_power(x)** = $\sqrt{x}$.

Solutie

```
> restart;
> fractional_power:=proc(x::{name,numeric})
    description "radical de ordinul n";
    local ind;
    if type(procname,`indexed`) # test if
  procedure has an index
      then ind:=op(procname);
      else ind:=2;
    end if;
    return x^(1/ind);
  end proc;
```

fractional_power := **proc**(x::{*name*, *numeric*}) **(15.1.1)**
local *ind*;
description "radical de ordinul n";
if *type*(*procname*, *indexed*) **then**
ind := *op*(*procname*)
else
ind := 2
end if;
return x^(1 / *ind*)
end proc

```
> fractional_power(2);
```

$$\sqrt{2} \tag{15.1.2}$$

```
> fractional_power[3](2);
```

$$2^{1/3} \tag{15.1.3}$$

```
> fractional_power[4](256);
```

$$256^{1/4} \tag{15.1.4}$$

```
> simplify(%);
```

$$4 \tag{15.1.5}$$

```
> fractional_power[n](x);
```

$$x^{\frac{1}{n}} \tag{15.1.6}$$

15. Indicii pot fi secvente. Scrieti o procedura `line` care are un argument x si pina la doi indici.
Iesirea lui `line` este dupa cum urmeaza: `line[a,b](x) = a + bx`, `line[a](x) = a[1] + a[2]x` si `line(x) = x`.

Solutie

```
> restart;
> line:=proc(x)
    description "procedura cu doi indici";
    local a,b;
    if type(procname,`indexed`) then # test if
  procedure has an index
       #print(procname,op(procname),nops
  (procname));
       if nops(procname)=1 then
           a:=op(procname);
           #print(a);
           return a[1]+a[2]*x;
       else
           a:=op(procname)[1];
           b:=op(procname)[2];
           return a+b*x;
       end if
    else
       return x;
    end if;
  end proc;
```

$line := \mathbf{proc}(x)$ **(16.1.1)**

local a, b;
description "procedura cu doi indici";
if $type(procname, indexed)$ **then**
if $nops(procname) = 1$ **then**
$a := op(procname)$; **return** $a[1] + a[2] * x$
else
$a := op(procname)[1]$; $b := op(procname)[2]$; **return** $a + b * x$
end if

```
    else
        return x
    end if
end proc
```

```
> line(x);
```

$$x \tag{16.1.2}$$

```
> line[a,b](x);
```

$$a + b\,x \tag{16.1.3}$$

```
> line[a](x);
```

$$a_1 + a_2\,x \tag{16.1.4}$$

16. Metoda secantei pentru determinarea unei radacini a ecuatiei $f(x) = 0$ se defineste prin

$$x_n = x_{n-1} - \frac{(x_{n-1} - x_{n-2})\, f(x_{n-1})}{f(x_{n-1}) - f(x_{n-2})}, \text{ pentru } 2 \le n.$$

Metoda secantei nu necesita derivate, dar necesita doua valori de pornire, x_0 si x_1. Pentru simplitate vom lua x_0 si x_1 numere aleatoare flotante generate prin `evalf(rand()/10^12)`.

(a) Scrieti o procedura Maple pentru a implementa formula de mai sus, adica a executa un pas al metodei secantei. Utilizati prototipul urmator:

```
secantstep := proc(f::procedure, x0::float,x1::float);
```

Testati pentru ecuatia $f(x) := \cos(x) - \frac{1}{2} = 0$.

(b) Utilizati `secantstep` pentru a defini o procedura Maple cu prototipul

```
secant1 := proc(f::procedure, n::nonnegint);
```

care returneaza x_n, pornind de la valorile aleatoare x_0 si x_1.

(c) Dati o implementare recursiva a metodei secantei, utilizand prototipul

```
secant2 := proc(f::procedure, n::nonnegint);
```

care returneaza x_n, pornind de la valorile aleatoare x_0 si x_1. Asigurati-va ca aceasta implementare recursiva este la fel de eficienta ca cea iterativa.

(a)

```
> restart;
> secantstep := proc(f::procedure,x0::float,
```

```
x1::float)
  description `un pas al metodei secantei`:
  local xn; #x nou, x precedent
  xn := x1-(x1-x0)/(eval(f(x1))-eval(f(x0)))*
eval(f(x1));
end proc;
```

secantstep := **proc**(*f*::*procedure*, *x0*::*float*, *x1*::*float*) **(17.1.1)**

local *xn*;

description `*un pas al metodei secantei*`;

$xn := x1 - (x1 - x0) * (eval(f(x1))) / (eval(f(x1)) - (eval(f(x0))))$

end proc

test executie

```
> Digits:=16;
```

$$Digits := 16 \tag{17.1.2}$$

```
> f := x -> cos(x) - 1/2;
```

$$f := x \rightarrow \cos(x) - \frac{1}{2} \tag{17.1.3}$$

```
> x0:=evalf(rand()/10^12);
```

$$x0 := 0.3957188605340000 \tag{17.1.4}$$

```
> x1:=evalf(rand()/10^12);
```

$$x1 := 0.1931398164150000 \tag{17.1.5}$$

```
> x1-(x1-x0)/(f(x1)-f(x0))*f(x1);
```

$$1.854893383005339 \tag{17.1.6}$$

```
> #x0:=0.5; x1:=1.5;
> x:=secantstep(f,x0,x1);
```

$$x := 1.854893383005339 \tag{17.1.7}$$

```
> evalf(Pi/3);
```

$$1.047197551196598 \tag{17.1.8}$$

Linia urmatoare se executa repetat

```
> x0:=x1: x1:=x: x:=secantstep(f,x0,x1); abs(x-
x1);
```

$$x := 1.047197551196598 \tag{17.1.9}$$

$$0.$$

(b)

```
> secant1 := proc(f::procedure,n::nonnegint)
    description "metoda secantei";
    local k,x0,x1,xn;
```

```
    x0:=evalf(rand()/10^12); x1:=evalf(rand()
/10^12);
    for k to n do
        xn:=secantstep(f,x0,x1);
        x0:=x1; x1:=xn;
    end do;
end proc;
```

$secant1 := \textbf{proc}(\,f::procedure, n::nonnegint)$ **(17.2.1)**

local $k, x0, x1, xn$;

description "metoda secantei";

$x0 := evalf(1/1000000000000 * rand(\,))$;

$x1 := evalf(1/1000000000000 * rand(\,))$;

for k **to** n **do**

$xn := secantstep(\,f, x0, x1);\ x0 := x1;\ x1 := xn$

end do

end proc

test executie

```
> st1:=time();
> sol:=secant1(f,8);
> time()-st1;
```

$$st1 := 0.828 \tag{17.2.2}$$

$$sol := Float(undefined)$$

$$0.$$

```
> f(sol);
```

$$-2.\ 10^{-16} \tag{17.2.3}$$

(c)

```
> secant2 := proc(f::procedure,n::nonnegint)
    description "metoda secantei, recursiv";
    local xn;
    global x0,x1;
    #print(n);
    xn:=x1-(x1-x0)/(eval(f(x1))-eval(f(x0)))*
eval(f(x1));
    x0:=x1; x1:=xn; #print(x0,x1);
    if (n=1) or abs(x1-x0)<10^(-Digits) then
      return xn;
    else
      xn:=secant2(f,n-1);
```

```
    end if
  end proc;
```

$secant2 := \mathbf{proc}(f::procedure, n::nonnegint)$ **(17.3.1)**

local xn;
global $x0, x1$;
description "metoda secantei, recursiv";
$xn := x1 - (x1 - x0) * (eval(f(x1)))$
$/ (eval(f(x1)) - (eval(f(x0))))$;
$x0 := x1$;
$x1 := xn$;
if $n = 1$ **or** $\mathrm{abs}(x1 - x0) < 10\text{\^{}}(-Digits)$ **then**
return xn
else
$xn := secant2(f, n - 1)$
end if
end proc

test executie

```
> x0:=evalf(rand()/10^12); x1:=evalf(rand()
  /10^12);

> st2:=time();
> secant2(f,8);
> time()-st2;
```

$$x0 := 0.5704134664770000 \quad \textbf{(17.3.2)}$$
$$x1 := 0.9920881460260000$$
$$st2 := 0.844$$
$$1.047197551196598$$
$$0.$$
$$0.844$$

```
> evalf(Pi/3);
```

$$1.047197551196598 \quad \textbf{(17.3.3)}$$

17. Executati `diff(sin(x),x);` si schimbati tabela remember a lui `diff` astfel ca la urmatoarea executie a lui `diff(sin(x),x);` sa obtinem sin(x).

Solutie

```
> restart;
> diff(sin(x),x);
```

$$\cos(x) \quad \textbf{(18.1.1)}$$

```
> T:=op(4,eval(diff));
```

$$T := table([\,(\sin(x), x) = \cos(x)\,])$$ **(18.1.2)**

```
> T[(sin(x),x)]:=sin(x);
```

$$T_{\sin(x),\,x} := \sin(x)$$ **(18.1.3)**

```
> diff(sin(x),x);
```

$$\sin(x)$$ **(18.1.4)**

```
> diff(exp(sin(x)),x);
```

$$\sin(x)\,e^{\sin(x)}$$ **(18.1.5)**

```
> forget(diff);
> diff(sin(x),x);
```

$$\cos(x)$$ **(18.1.6)**

18. Numerele lui Bell $B(n)$ se definesc prin $B(0) = 1$ si $B(n) = \sum_{i=0}^{n-1} \mathrm{binomial}(n-1, i)\, B(i)$, $0 < n$. Ele dau numarul de partitii ale unei multimi de n elemente. Scrieti o procedura recursiva pentru calculul lui $B(n)$. Asigurati-va ca procedura este suficient de eficienta pentru a calcula $B(50)$. Comanda `binomial(n,k)` calculeaza combinari de n luate cate k.

Solutie

```
> restart;
> BellNumber:=proc(n::nonnegint)::nonnegint;
     local i;
     description `compute Bell numbers`;
     option remember;
     if n=0 then
        1
     else
        add(binomial(n-1,i)*BellNumber(i),i=0..
  n-1);
     end if;
  end proc;
```

BellNumber := **proc**(*n*::*nonnegint*)::*nonnegint*; **(19.1.1)**
  option *remember*;
  local *i*;
  description `*compute Bell numbers*`;
  if $n = 0$ **then**
    1

```
    else
        add(binomial(n - 1, i) * BellNumber(i), i = 0 ..n - 1)
    end if
end proc
```

```
> BellNumber(0);
```

$$1 \tag{19.1.2}$$

```
> BellNumber(1);
```

$$1 \tag{19.1.3}$$

```
> BellNumber(5);
```

$$52 \tag{19.1.4}$$

```
> BellNumber(2);
```

$$2 \tag{19.1.5}$$

```
> BellNumber(3);
```

$$5 \tag{19.1.6}$$

Functia **bell** din pachetul **combinat** calculeaza numerele lui Bell. Iata aici niste exemple si implementarea

```
> combinat[bell](3);
```

$$5 \tag{19.1.7}$$

```
> combinat[bell](5);
```

$$52 \tag{19.1.8}$$

```
> interface(verboseproc=3);
```

$$1 \tag{19.1.9}$$

```
> print(combinat[bell]);
```

(19.1.10)

```
proc(n::algebraic)
    option remember, system;
    local bn, bc, r;
    if type(n, 'integer') then
        if n < 2 then
            1
        else
            bn, bc := 0, 1;
            for r from 0 to n - 1 do
                bn, bc := bn + bc * procname(r), iquo(bc
                * (n - r - 1), 1 + r)
            end do;
            bn
        end if
    else
        'procname(n)'
    end if
end proc
```

19. Polinomul Cebisev de grad n se poate defini ca fiind $\cos(n \arccos(x))$. Scrieti o procedura `C` care are un parametru de intrare x si un index n. Astfel, `C[n](x)` returneaza $\cos(n \arccos(x))$, iar `C[10](0.5)` returneaza valoarea polinomului Cebisev de grad 10 in punctul 0.5. Comparati aceasta valoare cu `orthopoly[T](10,0.5)`.

Solutie

```
> restart;
> C:=proc(x)
     description `polinomul Cebisev de grad n`;
     local n;
     if type(procname,indexed) then
        n:=op(procname);
     else n:=0;
     end if;
     expand(cos(n*arccos(x)));
  end proc;
```

C := **proc**(x)
  local n;
  description `polinomul Cebisev de grad n`;
  if $type(procname, indexed)$ **then**
    $n := op(procname)$
  else
    $n := 0$
  end if;
  $expand(\cos(n * \arccos(x)))$
end proc **(20.1.1)**

```
> C[2](t);
```

$$2 t^2 - 1 \quad \textbf{(20.1.2)}$$

```
> C[5](x);
```

$$16 x^5 - 20 x^3 + 5 x \quad \textbf{(20.1.3)}$$

$$2 t^2 - 1 \quad \textbf{(20.1.4)}$$

```
> C[10](t);
```

$$512 t^{10} - 1280 t^8 + 1120 t^6 - 400 t^4 + 50 t^2 - 1 \quad \textbf{(20.1.5)}$$

```
> C[10](0.5);
```

$$-0.5000000017 \quad \textbf{(20.1.6)}$$

```
> orthopoly[T](10,0.5);
```

$$-0.5000000000 \quad \textbf{(20.1.7)}$$

20. Fie `L[n](x)` un tip special de polinom Laguerre de grad n in variabila x. Definim `L[n](x)` prin `L[0](x) = 1`, `L[1](x) = 1-x` si pentru orice grad $n > 1$: `n*L[n](x) = (2*n-1-x)*L[n-1](x) - (n-1)*L[n-2](x)`. Scrieti o procedura Maple pentru calculul lui `L[n](x)`. Utilizati un index pentru gradul n si un parametru de intrare x. Asigurati-va ca procedura dumneavoastra poate calcula polinomul Laguerre de grad 50.

Solutie

```
> restart;
> L:=proc(x)
    description `polinomul Laguerre de grad n`;
    local n;
    if type(procname,indexed) then
       n:=op(procname);
    else n:=0;
    end if;
    if n=0 then
       1;
    elif n=1 then
       1-x;
    else
       expand((2*n-1-x)/n*L[n-1](x)-(n-1)/n*L
  [n-2](x));
    end if
  end proc;
```

$L :=$ **proc**(x) **(21.1.1)**

local n;
description `polinomul Laguerre de grad n`;
if $type(procname, indexed)$ **then**
  $n := op(procname)$
else
  $n := 0$
end if;
if $n = 0$ **then**
  1
elif $n = 1$ **then**
  $1 - x$
else
  $expand((2 * n - 1 - x) * L[n - 1](x) / n - (n - 1) * L$

$[n-2](x)/n)$

end if

end proc

```
> L[0](x);
```

$$1 \tag{21.1.2}$$

```
> L[1](x);
```

$$1-x \tag{21.1.3}$$

```
> L[2](x);
```

$$1-2x+\frac{1}{2}x^2 \tag{21.1.4}$$

```
> L[3](x);
```

$$1-3x+\frac{3}{2}x^2-\frac{1}{6}x^3 \tag{21.1.5}$$

```
> orthopoly[L](2,x);
```

$$1-2x+\frac{1}{2}x^2 \tag{21.1.6}$$

```
> orthopoly[L](3,x);
```

$$1-3x+\frac{3}{2}x^2-\frac{1}{6}x^3 \tag{21.1.7}$$

21. Introduceti `l := [seq(rand() mod 100,i=1..10)];` pentru a genera o lista de 10 numere aleatoare intre 0 si 99. Dati comenzile Maple pentru urmatoarele operatii
(a) impartiti fiecare element al listei la 100;
(b) convertiti lista intr-o lista de numere in virgula flotanta cu 3 cifre zecimale;
(c) selectati toate elementele listei > 0.5;
(d) calculati suma elementelor listei.

Solutie

```
> restart;
> l := [seq(rand() mod 100,i=1..10)];
```

$$l := [34, 15, 65, 59, 69, 42, 40, 80, 50, 65] \tag{22.1.1}$$

```
> #(a)
> l1:=map(x->x/100,l);
```

$$l1 := \left[\frac{17}{50}, \frac{3}{20}, \frac{13}{20}, \frac{59}{100}, \frac{69}{100}, \frac{21}{50}, \frac{2}{5}, \frac{4}{5}, \frac{1}{2}, \frac{13}{20}\right] \tag{22.1.2}$$

```
> l2:=map(x->evalf(x,3),l1);
```

$$l2 := [0.340, 0.150, 0.650, 0.590, 0.690, 0.420, 0.400, 0.800, 0.500, 0.650] \tag{22.1.3}$$

```
> select(x->(x>0.5),l2);
```

$$[0.650, 0.590, 0.690, 0.800, 0.650] \quad (22.1.4)$$

```
> add(l[i],i=1..nops(l));
```

$$519 \quad (22.1.5)$$

22. Introduceti comanda `p := randpoly([x,y],degree=7,dense);` pentru a genera un polinom aleator dens de gradul 7 in x si y. Dati comenzile Maple care elimina toti termenii de grad mai mic decat gradul lui `p`.

Solutie

```
> restart;
> p := randpoly([x,y],degree=7,dense);
```

$$p := 29 - 92\,x + 44\,y + 22\,x^6\,y - 94\,x^5\,y^2 + 87\,x^5\,y - 62\,x^4\,y^2 + 97\,x^4\,y - 4\,x^3\,y^4 - 83\,x^3\,y^3 - 10\,x^3\,y^2 + 62\,x^3\,y + 80\,x^2\,y^5 - 44\,x^2\,y^4 + 71\,x^2\,y^3 - 17\,x^2\,y^2 - 75\,x^2\,y - 7\,x^7 - 55\,x^6 + 87\,y^2 - 56\,x^5 - 23\,y^3 - 73\,x^4 + 37\,y^4 - 82\,x^3 + 72\,y^5 - 10\,x^2 + 74\,y^6 + 6\,y^7 - 7\,x\,y^6 - 40\,x\,y^5 + 42\,x\,y^4 - 50\,x\,y^3 + 23\,x\,y^2 + 75\,x\,y \quad (23.1.1)$$

```
> degree(p);
```

$$7 \quad (23.1.2)$$

```
> remove(x->(degree(x)<degree(p)),p);
```

$$22\,x^6\,y - 94\,x^5\,y^2 - 4\,x^3\,y^4 + 80\,x^2\,y^5 - 7\,x^7 + 6\,y^7 - 7\,x\,y^6 \quad (23.1.3)$$

23. Introduceti comanda `p := randpoly(x,degree=10,dense);` pentru a genera un polinom aleator dens de gradul 10 in x. Dati comenzile Maple care selecteaza acei termeni din `p` ai caror coeficienti sunt pozitivi.

Solutie

```
> restart;
> p := randpoly(x,degree=10,dense);
```

$$p := -4 - 7\,x^{10} + 22\,x^9 - 55\,x^8 - 94\,x^7 + 87\,x^6 - 56\,x^5 - 62\,x^3 + 97\,x^2 - 73\,x \quad (24.1.1)$$

```
> select(x->(op(1,x)>0),p);
```

$$22\,x^9 + 87\,x^6 + 97\,x^2 \quad (24.1.2)$$

24. Functia Zeta a lui Riemann se calculeaza cu comanda **Zeta(z)**, unde **z** este o expresie algebrica. Utilizati un macro pentru a defini **fzeta**, care ne da o aproximare in virgula flotanta (utilizand precizia standard din Digits) pentru functia zeta a lui Riemann. De exemplu, fzeta(3.0) da 1.202056903.

Solutie

```
> restart;
> macro(fzeta=evalf@Zeta);
```

$$fzeta \tag{25.1.1}$$

```
> fzeta(3.0);
```

$$1.202056903 \tag{25.1.2}$$

25. Presupunem ca scurgerea dintr-o conducta de petrol produce un cerc perfect pe podea. Aria cercului este πr^2, unde raza $r = r(t)$ este crescatoare in timp (in t). Utilizati compunerea a doua functii pentru a modela cresterea ariei cercului ca functie de timp, presupunand ca viteza cu care scurgerea apare este constanta.

Solutie

```
> restart;
> ecd:=diff(r(t),t)=c;
```

$$ecd := \frac{\mathrm{d}}{\mathrm{d}t}\, r(t) = c \tag{26.1.1}$$

```
> dsolve(ecd,r(t));
```

$$r(t) = c\, t + \_C1 \tag{26.1.2}$$

```
> r:=t->c*t;
```

$$r := t \rightarrow c\, t \tag{26.1.3}$$

```
> Aria:=r->Pi*r^2;
```

$$Aria := r \rightarrow \pi\, r^2 \tag{26.1.4}$$

```
> f:=Aria@r;
```

$$f := Aria@r \tag{26.1.5}$$

```
> f(t);
```

$$\pi\, c^2\, t^2 \tag{26.1.6}$$