

dl

Développements limités

Calculatrices : 89 92 92+ V200

Niveaux : Sup

Descriptif : Groupe de fonctions - Calculs de développements limités (quotient, composition, réciproque)

Auteur : Claude Morin

Mots-clefs : développements limités

Date de dernière révision : Septembre 2002

Présentation :

Il s'agit de deux groupes de 3 fonctions déterminant des développements limités :

- *dlexpr.** contient des fonctions agissant sur des expressions
 - *dllost.** contient les mêmes fonctions mais agissant sur des listes (ordre croissant)
- où * désigne à chaque fois 89g, 9xg ou v2g selon la calculatrice.

Ces trois fonctions sont les suivantes :

- *dlcom** : composition
- *dlquo** : quotient
- *dlrec** : réciproque

où * désigne à chaque fois soit *p* soit *l*.

Mode d'emploi :

Pour calculer le développement limité d'un quotient :

Il suffit de taper `dlquo(num(x),denom(x),x,n)` pour calculer un DL à l'ordre *n* du quotient $num(x)/denom(x)$.

Par exemple : `dlquo(x,exp(x)-1,x,3)` donne un DL à l'ordre 2 (simplification par *x*)

Pour calculer le développement limité d'une composée :

Il suffit de taper `dlcomp(f(x),g(x),x,n)` pour calculer un DL à l'ordre *n* de $f(g(x))$.

Par exemple : `dlcomp(sin(t),tan(t),t,7)` donne le DL à l'ordre 7 de $\sin(\tan(t))$;

Pour calculer le développement limité d'une réciproque :

Il suffit de taper `dlrecp(f(x),x,n)` pour calculer un DL à l'ordre *n* de la réciproque de $f(x)$.

Par exemple : `dlrecp(sin(t),t,5)` donne le DL de $\text{Arcsin}(t)$ à l'ordre 5.

Remarque : on aurait pu naturellement utiliser ici les versions "liste" des fonctions.

Sources :

DI coml (a,b,n)

Func

© dln de aob(x) (listes);entrer a,b,n

Local c,d,i,j,k

b→c:a[2]*b→d:d[1]+a[1]→d[1]

For k,2,n

For i,n,k,-1

$\Sigma(b[1+j]*c[1+i-j],j,1,i-k+1) \rightarrow c[1+i]$

EndFor

Ø→c[k]:d+a[1+k]*c→d

EndFor

d

EndFunc

DI comp (a,b,x,n)

Func

© dln de a(b(x));entrer a,b,x,n:utilise pcoefcn,dlcoml

pcoefcn(b,x,n)→b

a|x=b[1]+xx→a:Ø→b[1]

pcoefcn(a,xx,n)→a

expand(sum(dlcoml(a,b,n)*seq(x^k,k,Ø,n))+o(x^n))

EndFunc

DI quol (a,b,n)

Func

© dln de a/b (listes):entrer a,b,n

Local k,i,c

newList(n+1)→c

For k,Ø,n

If string(a[1+k])≠"Ø" Then

a[1+k]/(b[1])→c[1+k]

For i,k,n

a[1+i]-c[1+k]*b[1+i-k]→a[1+i]

EndFor

EndIf

EndFor

c

EndFunc

DI quop (a,b,x,n)

Func

© dln de a(x)/b(x);entrer a,b,x,n;utilise pcoefcn,dlquol

Local p,k,s

pcoefcn(a,x,n)→a:pcoefcn(b,x,n)→b:Ø→p

While string(b[p+1])="Ø"

p+1→p

EndWhile

$\Sigma(abs(a[k]),k,1,p) \rightarrow s$

If string(s)≠"Ø" Then

"impossible"

Else

n-p→n

$\text{right}(a, n+1) \rightarrow a : \text{right}(b, n+1) \rightarrow b$
<pre> expand(sum(dlquo1(a,b,n)*seq(x^k,k,0,n))+o(x^n)) EndIf EndFunc DI recl (a,n) Func © dln réciproque de a (a0=0); entrer {a1,...},n Local b,c,d,i,j,k a→c:newList(n)→b:1/(a[1])→b[1] ~b[1]*a→d:1+d[1]→d[1] For k,2,n For i,n,k,-1 Σ(a[j]*c[i-j],j,1,i-k+1)→c[i] EndFor 0→c[k-1] d[k]/(c[k])→b[k]:d-b[k]*c→d EndFor DI recp (a,x,n) Func © dln réciproque de a(x) (a0=0); entrer a,x,n; utilise pcoefcn,dlrecl pcoefcn(a,x,n)→a If string(a[1])≠"0" Return "erreur:a0≠0" expand(sum(dlrecl(right(a,n),n)*seq(x^k,k,1,n))+o(x^n)) EndFunc </pre>