Function: GTP:-`type/gradedmonom` - define a type 'gradedmonom'

Calling Sequence:

type(p,gradedmonom);

Parameters:

p - element of one of these types: `*`, function, algebraic.

Description:

- Monomial (homogeneous) elements in the graded tensor product Cl(B1) &t Cl(B2) &t ... &t Cl(Br) of r Clifford algebras Cl(Bi), where Bi are quadratic forms, 1<=i<=r, are by definition of type *'gradedmonom'*. Thus, they are either of type Clifford:-`type/tensorprod` or they are products of two elements, one of type Clifford:-`type/tensorprod` and one of type Clifford:-`type/cliscalar`.
- See also GTP:-`type/gradedpolynom`.

Examples:

```
> restart:with(Clifford):with(GTP):
> type(e1 &t e1,gradedmonom),type(Pi*(e1we2 &t e1 &t
  e2),gradedmonom);
Cliplus has been loaded. Definitions for type/climon and type/clipolynom now in
clude &C and &C[K]. Type ?cliprod for help.
```

true, *true*

```
> type(2*&t(e1,e2,e3),gradedmonom);
```

true

```
> type(2*&t(e1,e2,e3)+e2we3 &t e2we1,gradedmonom);
```

false

```
> type(2*&t(e1,e2,e3)+e2we3 &t e2we1,gradedpolynom);
```

true

```
> dim:=2:K:=cbasis(2):L:=gbasis(K$2);
```

L := [*Id* &t *Id*, *Id* &t *e1*, *Id* &t *e2*, *Id* &t *e1we2*, *e1* &t *Id*, *e1* &t *e1*, *e1* &t *e2*, *e1* &t *e1we2*, *e2* &t *Id*, *e2* &t *e1*, *e2* &t *e2*, *e2* &t *e1we2*, *e1we2* &t *Id*, *e1we2* &t *e1*, *e1we2* &t *e2*, *e1we2* &t *e1we2*]

```
> map(type,L,gradedmonom);
```

[*true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*]

```
>
>
```

See Also: GTP:-gbasis, GTP:-`type/gradedodd`, GTP:-grade, GTP:-`&t`, Clifford:-`type/tensorprod`, GTP:-gradedprod, GTP:-gprod, GTP:-`type/gradedeven`

Last revised: December 20, 2008 /RA

Function: GTP:-`type/gradedodd` - define type 'gradedodd'

Calling Sequence:

type(p,gradedodd);

Parameters:

p - element of one of these types: GTP:-`type/gradedpolynom`, GTP:-`type/gradedmonom`, Clifford:-`type/tensorprod`

Description:

- Polynomial elements in the graded tensor product Cl(B1) &t Cl(B2) &t ... &t &t Cl(Br) of r Clifford algebras Cl(Bi), where Bi are quadratic forms, 1<=i<=r, which contain all monomials of grade 1 are of type *'gradedodd'*.
- Polynomials of type 'gradedodd' do not form a subalgebra in the graded tensor algebra Cl(B1) &t Cl(B2) &t ... &t Cl(Br). However, a product of two odd elements is even.
- See also GTP:-`type/gradedeven` and GTP:-grade.

Examples:

```
> restart:with(Clifford):with(GTP):
> p1:=e1 &t e1 &t e2+ 2* (e1 &t e2 &t e2);
Cliplus has been loaded. Definitions for type/climon and type/clipolynom now in
clude &C and &C[K]. Type ?cliprod for help.
```

$$p1 := ((e1 \,\&t\, e1) \,\&t\, e2) + 2\,((e1 \,\&t\, e2) \,\&t\, e2)$$

```
> type(p1,gradedeven),type(p1,gradedodd);
```

false, *true*

```
> p2:=2*&t(e1,e2,e3);type(p2,gradedodd);
```

$$p2 := 2\,((e1 \,\&t\, e2) \,\&t\, e3)$$

true

```
> B:=linalg[diag](1,1,1):type(gradedprod(p1,p2),gradedeven);
```

true

```
> dim:=2:K:=cbasis(2):L:=gbasis(K$2);
```

$L := [Id \,\&t\, Id, Id \,\&t\, e1, Id \,\&t\, e2, Id \,\&t\, e1we2, e1 \,\&t\, Id, e1 \,\&t\, e1, e1 \,\&t\, e2, e1 \,\&t\, e1we2,$
$e2 \,\&t\, Id, e2 \,\&t\, e1, e2 \,\&t\, e2, e2 \,\&t\, e1we2, e1we2 \,\&t\, Id, e1we2 \,\&t\, e1, e1we2 \,\&t\, e2,$
$e1we2 \,\&t\, e1we2]$

```
> map(type,L,gradedodd);
```

[*false*, *true*, *true*, *false*, *true*, *false*, *false*, *true*, *true*, *false*, *false*, *true*, *false*, *true*, *true*, *false*]

```
> map(type,L,gradedeven);
```

[*true*, *false*, *false*, *true*, *false*, *true*, *true*, *false*, *false*, *true*, *true*, *false*, *true*, *false*, *false*, *true*]

```
>
>
```

See Also: GTP:-`type/gradedmonom`, GTP:-gbasis, GTP:-grade, GTP:-`&t`, Clifford:-`type/tensorprod`, GTP:-gradedprod, GTP:-gprod, GTP:-`type/gradedeven`

Last revised: December 20, 2008 /RA

Function: GTP:-`type/gradedpolynom` - define a type 'gradedpolynom'

Calling Sequence:

type(p,gradedpolynom);

Parameters:

p - element of one of these types: `*`, `+`, 'function', or 'algebraic'.

Description:

- Polynomial elements in the graded tensor product Cl(B1) &t Cl(B2) &t ... &t &t Cl(Br) of r Clifford algebras Cl(Bi), where Bi are quadratic forms, 1<=i<=r, are by definition of type *'gradedpolynom'*. Thus, they are linear combinations of the basis elements of the type Clifford:-`type/tensorprod` while their coefficients are of the type Clifford:-`type/cliscalar`.
- See also GTP:-`type/gradedmonom`. Elements of type 'gradedmonom' are also of type 'gradedpolynom'.
- Elements of the type 'gradedpolynom' are multivariate polynomials used by the procedures GTP:-gprod and GTP:-gradedprod.

Examples:

```
> restart:with(Clifford):with(GTP):
> type(e1 &t e1 + 2* (e1we2 &t e2),gradedpolynom),
  type(Pi*(e1we2 &t e1 &t e2),gradedpolynom);
Cliplus has been loaded. Definitions for type/climon and type/clipolynom now in
clude &C and &C[K]. Type ?cliprod for help.
```

true, *true*

```
> type(2*&t(e1,e2,e3),gradedpolynom);
```

true

```
> dim:=2:K:=cbasis(2):L:=gbasis(K$2);
```

$L := [Id \&t\ Id, Id \&t\ e1, Id \&t\ e2, Id \&t\ e1we2, e1 \&t\ Id, e1 \&t\ e1, e1 \&t\ e2, e1 \&t\ e1we2, e2 \&t\ Id, e2 \&t\ e1, e2 \&t\ e2, e2 \&t\ e1we2, e1we2 \&t\ Id, e1we2 \&t\ e1, e1we2 \&t\ e2, e1we2 \&t\ e1we2]$

```
> map(type,L,gradedpolynom);
```

[*true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*, *true*]

```
>
>
```

See Also: GTP:-`type/gradedmonom`, GTP:-gbasis, GTP:-`type/gradedodd`, GTP:-grade, GTP:-`&t`, Clifford:-`type/tensorprod`, GTP:-gradedprod, GTP:-gprod, GTP:-`type/gradedeven`

Last revised: December 20, 2008 /RA

Function: GTP:-gradedprod - compute a graded product of elements of the type 'tensorprod', 'gradedmonom', or 'gradedpolynom'

Calling Sequence:

gradedprod(p1,p2);
gradedprod(p1,p2,B1,B2,...,Br);

Parameters:

p1, p2 - graded polynomials of type GTP:-`type/gradedpolynom` of rank r
B1,B2,...,Br - (optional) sequence of r diagonal forms Bi, 1<=i<=r, where r is the tensor rank of p1 and p2

Description:

- Procedure 'gradedprod' is an extension of GTP:-prod and computes a product of two graded polynomials in the graded tensor product Cl(B1) &t Cl(B2) &t ... &t Cl(Br) of r Clifford algebras Cl(B1), Cl(B2), ... , Cl(Br). In particular, it can handle also monomials.
- When the optional sequence is used, Clifford products are computed component-wise on homogeneous elements in Cl(B1), Cl(B2), ... , Cl(Br) with a help of the procedure GTP:-cmulB. However, the Z2-gradation is taken into consideration in order to assure, for example, that elements of the type e1 &t 1 and 1 &t e1 belonging to Cl(B1) &t Cl(B2) anticommute.
- For more information how multiplication is defined on homogeneous elements in the graded tensor product Cl(B1) &t Cl(B2) of two Clifford algebras Cl(B1) and Cl(B2) see GTP:-gprod. This definition is extended to tensors of higher ranks and then to non-homogeneous tensors by linearity.
- When the optional sequence is not used, the default bilinear form B is applied. Thus, in this case, the r products will be computed in r different copies of Cl(B).
- The ranks of p1 and p2 must be the same. They can be found with GTP:-tensorrank.

Examples:

```
> restart:with(Clifford):with(GTP):eval(makealiases(5)):_prolevel
  :=true:
```

Example 1:

```
> B:=linalg[diag](1,-1,-1,-1):
> type(e1 &t e2,tensorprod),
  type(2*(e1 &t e3),gradedmonom),
  type(2*&t(e1,e2,e3),gradedmonom);
  type((e1 &t e2) + b*(e1 &t e2) +e2 &t e3,gradedpolynom);
Cliplus has been loaded. Definitions for type/climon and type/clipolynom now in
clude &C and &C[K]. Type ?cliprod for help.
```

$$true, true, true$$

$$true$$

```
> gradedprod(e1 &t e2 &t e2,e1 &t e2 &t e2);tensorrank(%);
```
$$-((Id \mathbin{\&t} Id) \mathbin{\&t} Id)$$
$$3$$
```
> gradedprod((e1 + e2) &t e2 &t Id,e1 &t e2 &t Id);tensorrank(%);
```
$$((Id \mathbin{\&t} Id) \mathbin{\&t} Id) - ((e12 \mathbin{\&t} Id) \mathbin{\&t} Id)$$
$$3$$
```
> p:=a*(Id &t Id &t e2)-3*(e1we2 &t e2 &t e2we1 )+b*4*(Id &t e2
  &t e1)-
        Id &t e1we2 &t e1we2;
```
$$p := a\,((Id \mathbin{\&t} Id) \mathbin{\&t} e2) - 3\,((e12 \mathbin{\&t} e2) \mathbin{\&t} e21) + 4\,b\,((Id \mathbin{\&t} e2) \mathbin{\&t} e1) - ((Id \mathbin{\&t} e12) \mathbin{\&t} e12)$$
```
> type(p,gradedpolynom);tensorrank(p);
```
$$true$$
$$3$$
```
> r1:=gradedprod(-a*p,2*p);
```
$$r1 := 2\,a^3\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) + 12\,a^2\,((e12 \mathbin{\&t} e2) \mathbin{\&t} e1) - 16\,a^2\,b\,((Id \mathbin{\&t} e2) \mathbin{\&t} e12) + 16\,a\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) - 48\,a\,b\,((e12 \mathbin{\&t} Id) \mathbin{\&t} e2) - 32\,a\,b^2\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) + 16\,a\,b\,((Id \mathbin{\&t} e1) \mathbin{\&t} e2)$$
```
> tensorrank(r1);
```
$$3$$

The results from 'gradedprod' may be collected using GTP:-gcollect:
```
> gcollect(r1);
```
$$2\,a\,(8 + a^2 - 16\,b^2)\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) - 48\,a\,b\,((e12 \mathbin{\&t} Id) \mathbin{\&t} e2) + 12\,a^2\,((e12 \mathbin{\&t} e2) \mathbin{\&t} e1) - 16\,a^2\,b\,((Id \mathbin{\&t} e2) \mathbin{\&t} e12) + 16\,a\,b\,((Id \mathbin{\&t} e1) \mathbin{\&t} e2)$$

Example 2: In Example 1, r1 is an element of C(B) &t Cl(B) &t Cl(B) which is Z2-isomorphic with Cl(3B) where 3B is a quadratic form of signature (3,9). Now use different quadratic forms in each component. For example, consider p defined above as an element of the algebra Cl(B) &t Cl(B1) &t Cl(B2) where B1 and B2 are defined below.
```
> B1:=linalg[diag](-1,1,-1,1):B2:=linalg[diag](1,1,-1,1):
```
Now, we compute the graded product of p with p. Notice, that the result r2 is different than r1:
```
> r2:=gradedprod(p,p,B,B1,B2);
```
$$r2 := a^2\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) + 6\,a\,((e12 \mathbin{\&t} e2) \mathbin{\&t} e1) + 8\,a\,b\,((Id \mathbin{\&t} e2) \mathbin{\&t} e12) - 10\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) - 24\,b\,((e12 \mathbin{\&t} Id) \mathbin{\&t} e2) - 16\,b^2\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) + 8\,b\,((Id \mathbin{\&t} e1) \mathbin{\&t} e2)$$
```
> gcollect(r2);
```
$$(-10 + a^2 - 16\,b^2)\,((Id \mathbin{\&t} Id) \mathbin{\&t} Id) - 24\,b\,((e12 \mathbin{\&t} Id) \mathbin{\&t} e2) + 6\,a\,((e12 \mathbin{\&t} e2) \mathbin{\&t} e1) + 8\,a\,b\,((Id \mathbin{\&t} e2) \mathbin{\&t} e12) + 8\,b\,((Id \mathbin{\&t} e1) \mathbin{\&t} e2)$$

Example 3: Some more computations.

```
> p:=gradedprod(e1 &t e2,e2 &t e1we2-2*Pi*cos(alpha)*e1we2 &t Id
  -(a+b)/(a-b)*e2we1 &t e1);
```

$$p := -(e12 \,\&t\, e1) - 2\,\pi\cos(\alpha)\,(e2 \,\&t\, e2) - \frac{a\,(e2 \,\&t\, e12)}{a-b} - \frac{b\,(e2 \,\&t\, e12)}{a-b}$$

```
> gcollect(p);
```

$$-2\,\pi\cos(\alpha)\,(e2 \,\&t\, e2) - \frac{(a+b)\,(e2 \,\&t\, e12)}{a-b} - (e12 \,\&t\, e1)$$

```
> p1:=&t(-Pi*cos(alpha)*e1,e2-e2we1,Id,2*e1);
  p2:=&t(Id,e1-3*(a-b)/(a+c)*e1we2,e2we1,1);
```

$$p1 := -2\,\pi\cos(\alpha)\,(((e1 \,\&t\, e2) \,\&t\, Id) \,\&t\, e1) + 2\,\pi\cos(\alpha)\,(((e1 \,\&t\, e21) \,\&t\, Id) \,\&t\, e1)$$

$$p2 := (((Id \,\&t\, e1) \,\&t\, e21) \,\&t\, 1) - \frac{3\,(a-b)\,(((Id \,\&t\, e12) \,\&t\, e21) \,\&t\, 1)}{a+c}$$

```
> tensorrank(p1),tensorrank(p2);
```

$$4,\ 4$$

```
> r3:=gradedprod(p1,p2,B2,B1,B,B2);
```

$$r3 := 2\,\pi\cos(\alpha)\,(((e1 \,\&t\, e12) \,\&t\, e12) \,\&t\, e1) + \frac{6\,\pi\cos(\alpha)\,a\,(((e1 \,\&t\, e1) \,\&t\, e12) \,\&t\, e1)}{a+c} - \frac{6\,\pi\cos(\alpha)\,b\,(((e1 \,\&t\, e1) \,\&t\, e12) \,\&t\, e1)}{a+c} - 2\,\pi\cos(\alpha)\,(((e1 \,\&t\, e2) \,\&t\, e12) \,\&t\, e1) - \frac{6\,\pi\cos(\alpha)\,a\,(((e1 \,\&t\, Id) \,\&t\, e12) \,\&t\, e1)}{a+c} + \frac{6\,\pi\cos(\alpha)\,b\,(((e1 \,\&t\, Id) \,\&t\, e12) \,\&t\, e1)}{a+c}$$

```
> gcollect(r3);
```

$$-\frac{6\,\pi\cos(\alpha)\,(a-b)\,(((e1 \,\&t\, Id) \,\&t\, e12) \,\&t\, e1)}{a+c} + \frac{6\,\pi\cos(\alpha)\,(a-b)\,(((e1 \,\&t\, e1) \,\&t\, e12) \,\&t\, e1)}{a+c} + 2\,\pi\cos(\alpha)\,(((e1 \,\&t\, e12) \,\&t\, e12) \,\&t\, e1) - 2\,\pi\cos(\alpha)\,(((e1 \,\&t\, e2) \,\&t\, e12) \,\&t\, e1)$$

Thus, element r3 belongs to an algebra Z2-isomorphic with Cl(B2,B1,B,B2), that is, Cl(B3) where B3 is a quadratic form of signature (9,7).

```
>
>
```

See Also: GTP:-`type/gradedmonom`, GTP:-gbasis, GTP:-`type/gradedodd`, GTP:-grade, GTP:-`&t`, Clifford:-`type/tensorprod`, GTP:-`type/tensorrank`, GTP:-`type/gradedeven`

Last revised: December 20, 2008 /RA

Function: GTP:-tensorrank - find the rank of a tensor or an elements of the type 'gradedpolynom'

Calling Sequence:

tensorrank(p);

Parameters:

p - graded polynomials of type GTP:-`type/gradedpolynom`

Description:

- Procedure 'tensorrank' finds the rank of a graded polynomial, that is, an element of the graded tensor product Cl(B1) &t Cl(B2) &t ... &t Cl(Br) of some Clifford algebras Cl(B1), Cl(B2), ... , Cl(Br).
- This procedure is needed to determine if the tensors entered in GTP:-gprod or GTP:-gradedprod are of the same rank.
- If, by mistake, tensors of different ranks are detected in p, an error message is returned.

Examples:

```
> restart:with(Clifford):with(GTP):
> tensorrank(e1 &t e2);
Cliplus has been loaded. Definitions for type/climon and type/clipolynom now in
clude &C and &C[K]. Type ?cliprod for help.
```

$$2$$

```
> tensorrank(&t((e1 + e2),e2,Id,e1+e2,Id));
```

$$5$$

```
> p:=a*(Id &t Id &t e2)-3*(e1we2 &t e2 &t e2we1 )+b*4*(Id &t e2
  &t e1)-
      Id &t e1we2 &t e1we2;
```

$$p := a\,((Id \,\&t\, Id) \,\&t\, e2) - 3\,((e1we2 \,\&t\, e2) \,\&t\, e2we1) + 4\,b\,((Id \,\&t\, e2) \,\&t\, e1) - ((Id \,\&t\, e1we2) \,\&t\, e1we2)$$

```
> tensorrank(p);
```

$$3$$

```
> p:=&t(e1,e2,e3)-&t(e1,e2);
```

$$p := ((e1 \,\&t\, e2) \,\&t\, e3) - (e1 \,\&t\, e2)$$

```
> tensorrank(p); #testing an error message
Error, (in GTP:-tensorrank) tensors of mixed ranks are found in
&t(&t(e1,e2),e3)-&t(e1,e2)
>
>
```

See Also: GTP:-`type/gradedmonom`, GTP:-gbasis, GTP:-`type/gradedodd`, GTP:-grade, GTP:-`&t`, Clifford:-`type/tensorprod`, GTP:-gradedprod, GTP:-`type/gradedeven`

Last revised: December 20, 2008 /RA