# 13. Substitution, expansion and factorization

## 13.1 Substitution

In the previous lecture, we have seen how to factor a cosine polynomial with the aid of subs.

Another application of the substitute command is to control the simplification of expressions. Consider the following example:

```
[> p := (x+y)^2 + 1/(x+y)^2;
[> normal(p);
```

A better representation of p is to have $(x+y)^4+1$ in the numerator. We can achieve this by replacing $x+y$ by some variable $z$, normalize the substituted expression and the replace $z$ by $x+y$ after the normalization.

```
[> pz := subs(x+y=z,p);
```

Observe the syntax of the command: in the equality we substitute left hand side by right hand side anywhere the left hand side occurs in the second argument of p.

```
[> npz := normal(pz);
[> q := subs(z=x+y,npz);
```

With **subsop** we have a more subtle control over substitution. Suppose we wish to replace $x+y$ by $x-y$ in the denominator only.

```
[> op(2,q); op([2,1],q);
```

With the above op commands we figured out that the denominator is the first element of second operand sequence of q, denoted by [2,1] (reading from left to right). To replace $x+y$ by $x-y$, we apply the subsop command as follows:

```
[> subsop([2,1]=x-y,q);
```

In multivariate expressions, the order of substitution matters. We distinguish sequential and simultaneous substitution.

```
[> p := a + 2*b + 3*c;
```

Suppose we wished to permute the variables in a cyclic way, a becomes b, b becomes c and c becomes a :

```
[> subs(a=b,b=c,c=a,p);
```

The above substitute executes the substitutions one after the other, which is equivalent to

```
[> subs(c=a,subs(b=c,subs(a=b,p)));
```

But in a cyclic permutation, we want the substitutions done simultaneously, all at once:

```
[> subs({a=b,b=c,c=a},p);
```

From your earlier algebra courses, you may remember that substitution is one way to solve linear equations and to simplify expressions.

Suppose we new that `a+b = c`, then we could simplify the expression for p above. We can try with subs:

```
[> subs(a+b=c,p);
```

This does not work, since subs does not recognize the algebraic structure. Instead we must use the algebraic substitution command:

```
[> algsubs(a+b=c,p);
```

The subs is a syntactical substitution, it only substitutes when there is an exact match for a+b, ignoring the mathematical structures.

Simplifying general multivariate polynomials with respect to side relations is a hard problem.

## 13.2 Expansion

We have already seen that Maple does not expand automatically. To expand we have the expand and Expand commands. For example, if we multpliy $a + b + c$ with some polynomial:

```
[> p := (a+b+c)*(x^3 +9*x + 8);
[> expand(p);
```

But suppose we wished to expand only partially, leaving the polynomial intact:

```
[> op(2,p);
[> q := subsop(2=z,p);
[> eq := expand(q);
[> ep := subs(z=op(2,p),eq);
```

For algebraic numbers, we use the Expand command, mainly in conjunction with modulo, when working over finite fields. The experiment below, shows what needs to be done when extending the rationals with a root of an irreducible polynomial:

```
[> p2 := op(2,p);
[> irreduc(p2);
[> alias(alpha=RootOf(p2));
```

Suppose we now calculate in $\mathbb{Q}(\alpha) = \{ a + b\alpha + c\alpha^2 \mid a, b, c \in \mathbb{Q} \}$.

```
[> ap := (alpha+1)^7;
```

Usually we would not wish to expand an expression as the above, but it simplifies because alpha is a root of a cubic polynomial:

```
[> expand(ap);
[> evala(%);
```

Of course, we had already seen the simplify command :

```
[> simplify(ap);
```

## 13.3 Factorization

We have already seen the difference between factor and Factor, and in particular the importance of the number field.

```
[> p := x^2 + 2;
[> factor(p);
```

This polynomial does not factor over the rational numbers, but we can extend the rational numbers with the square root of two and the imaginary unit :

```
[> factor(p,{sqrt(2),I});
```

Now we knew what we needed to add as extensions to the factor command. In most cases we will not have this information, therefore, Maple offers the command split as part of the polytools package (new since Maple 7):

```
[> fp := polytools[split](p,x);
```

At first it seems as if Maple has not done much, but you can convert to a nicer representation:

```
[> rfp := convert(fp,radical);
```

To see the imaginary unit appearing, we apply evalc (evaluate complex). Simply applying evalc to rfp would expand the polynomial, something we do not want. Instead we apply evalc to all the operands of the expression rfp, for which the command map is appropriate:

```
[> map(evalc,rfp);
```

## 13.4 Assignments

1. Give the Maple commands to transform
$$(x+y)^2 + \frac{1}{x+y}$$
into
$$\frac{(x+y)^3+1}{x+y}$$
and vice versa.

2. Give the Maple commands to transform
$$x^2+2x+1+\frac{1}{x^2+2x+1}$$
into
$$\frac{(x+1)^4+1}{(x+1)^2}$$
and vice versa.

3. Give the Maple commands to transform
$$x^3 - xy^2 - yx^2 + y^3 + x^2 - y^2$$
into
$$(x^2-y^2)(x-y+1).$$

4. Give the Maple commands to transform
$$(x+z^2+1)(y-z^2-1)$$
into
$$xy - x(z^2+1) + (z^2+1)y - (z^2+1)^2.$$

5. Consider the polynomial $p = 92 - 93x - 8x^2$.
Give the Maple command(s) to write $p$ as
$$-8\left(x+\frac{93}{16}-\frac{\sqrt{11593}}{16}\right)\left(x+\frac{93}{16}-\frac{\sqrt{11593}}{16}\right).$$