

# Introduction à Maple - correction

Nicolas Gast - nicolas.gast@ens.fr

6 février 2006

## 1 Correction de l'exercice 2 :

*Montrer que*

$$\arctan(a) = \arctan(b) + \arctan\left(\frac{a-b}{1+ab}\right)$$

Maple ne sait pas résoudre ce problème en utilisant uniquement la commande `simplify`. Pour montrer ce résultat, il faut donc ruser. On peut par exemple développer  $\tan(\arctan(b) + \arctan((a-b)/(1+ab)))$  donnant un résultat assez indigeste, on peut demander à Maple de le factoriser.

```
> factor( expand( tan ( arctan(b) + arctan( (a-b)/(1+a*b) ) ) ) );  
a
```

Ce calcul montre que  $\arctan(a) = \arctan(b) + \arctan\left(\frac{a-b}{1+ab}\right) + k\pi (k \in \mathbb{Z})$ . Je vous laisse démontrer ensuite que  $\forall (a, b) \in \mathbb{R} : k = 0$

*Remarque* : la notion de simplification est de factorisation n'est pas toujours évidente pour un programme informatique. En particulier lorsque des fonctions assez complexes interviennent dans les formules comme les fonctions trigonométriques, deux êtres humains ne seront pas toujours d'accord sur laquelle de deux formules équivalentes est la plus simple. Il faut donc souvent ruser pour arriver à montrer certaines formules avec Maple. Souvent Maple ne fournit pas une démonstration mais aide aux calculs.

## 2 Correction de l'exercice 5 :

*Implémenter l'algorithme d'Euclide vu plus haut.*

Implémenter un algorithme revient à traduire l'algorithme écrit en français en un langage compréhensible par l'ordinateur afin d'écrire une fonction (une *procedure* en Maple) qui résout le problème.

En Maple, une fonction ressemble à ça :

```
ma_fonction := proc (argument_1, argument_2, ... , argument_n)  
  # ici se trouvent les commandes  
  # généralement sur plusieurs lignes  
end;
```

Ensuite il s'agit de connaître la traduction des commandes en Maple.

- Le reste de la division euclidienne de  $a$  par  $b$  s'écrit  $a \bmod b$ .
- La commande `si cond alors a sinon b finsi` s'écrit `if cond then a; else b; fi;`
- Mettre la valeur  $x$  dans  $r$  s'écrit `t := x;`

Les sauts de lignes (shift-enter) ainsi que les espaces sont ignorés par Maple, n'hésitez donc pas à en abuser afin de rendre votre programme plus lisible. L'algorithme d'Euclide peut s'implémenter en Maple par :

```
PGCD := proc(a,b)  
  local r; #cette ligne indique à maple que la variable r sera utilisée dans le programme  
  if a<b  
    then PGCD(b,a)  
    else if b=0  
      then a;  
      else r := a mod b;  
           PGCD(b,r);  
      fi;  
    fi;  
  fi;  
end;
```