

## Corrigé TP9 - Fonctions

### Exercice 09.1

```
PROGRAM somme_de_tangentes ;
VAR n,k : INTEGER ;
    s : REAL ;
FUNCTION tan(x : REAL) : REAL ;
    BEGIN
        tan := (sin(x))/(cos(x)) ;
    END ;
BEGIN
    WRITELN('Donner un entier n') ;
    READLN(n) ;
    s := 0 ;
    FOR k:=1 TO n
        DO s := s + tan(k/n) ;
    WRITELN('La somme vaut : ',s) ;
    READLN ;
END.
```

### Exercice 09.2

REMARQUE : à chaque fois qu'on calcule des factorielle, on les enregistre dans des cases de type REAL, (sinon cela dépasse vite la capacité d'une case INTEGER, et cela donne n'importe quoi).

1. Si on veut calculer le produit  $n! = 1 \times 2 \times \dots \times n$ , il suffit de faire une petite boucle For.

```
PROGRAM calcul_de_factorielle ;
VAR n,k : INTEGER;
    p : REAL ;
BEGIN
    WRITELN('Donner un entier n') ;
    READLN(n) ;
    p := 1 ;
    FOR k := 2 TO n
        DO p := p*k ;
    WRITELN(n,'! vaut ',p) ;
    READLN ;
END.
```

```

2. PROGRAM calcul_de_factorielle_avec_la_fonction ;
   VAR n : INTEGER;
   FUNCTION fact(x : INTEGER) : REAL ;
     VAR k : INTEGER ;
     prod : REAL ;
     BEGIN
       prod := 1 ;
       FOR k := 2 TO n
         DO prod := prod*k ;
       fact := prod ;
     END ;
   BEGIN
     WRITELN('Donner un entier n') ;
     READLN(n) ;
     WRITELN(n,'! vaut ',fact(n)) ;
     READLN ;
   END.

3. PROGRAM calcul_de_binomiaux_avec_la_fonction ;
   VAR n,p : INTEGER;
   FUNCTION fact(x : INTEGER) : REAL ;
     VAR k : INTEGER ;
     prod : REAL ;
     BEGIN
       prod := 1 ;
       FOR k := 2 TO n
         DO prod := prod*k ;
       fact := prod ;
     END ;
   BEGIN
     WRITELN('Donner un entier n') ;
     READLN(n) ;
     WRITELN('Donner un entier p tel que 0 <= p <= n') ;
     READLN(p) ;
     WRITELN(p,' parmi ',n,' vaut ',fact(n)/(fact(p)*fact(n-p))) ;
     READLN ;
   END.

4. PROGRAM affiche_tous_les_factorielles ;
   VAR n,i : INTEGER;
   FUNCTION fact(x : INTEGER) : REAL ;
     VAR k : INTEGER ;
     prod : REAL ;
     BEGIN
       prod := 1 ;
       FOR k := 2 TO n
         DO prod := prod*k ;
       fact := prod ;
     END ;
   BEGIN
     WRITELN('Donner un entier n') ;
     READLN(n) ;
     FOR i := 0 TO n
       DO WRITELN(i,'! vaut ',fact(i)) ;
     READLN ;
   END.

```

```

5.      PROGRAM affiche_tous_les_factorielles ;
        VAR n,i : INTEGER;
            prod : REAL ;
        BEGIN
            WRITELN('Donner un entier n') ;
            READLN(n) ;
            prod := 1 ;
            WRITELN(0! vaut 1) ;
            FOR i := 1 TO n
                DO
                    BEGIN
                        prod := prod*i ;
                        WRITELN(i,'! vaut ',prod) ;
                    END ;
                READLN ;
            END.

```

Dans ce programme, l'ordinateur fait moins de calculs que dans le précédent, car il réutilise à chaque fois le calcul précédent. La fonction factorielle recommence le calcul de  $n!$  au début à chaque étape.