

# 1 - Règles de grammaire

$\langle \text{script} \rangle ::= \text{'SCRIPT'} \{ \langle \text{instruction} \rangle \} \text{'FIN'}$

$\langle \text{instruction} \rangle ::= \langle \text{action} \rangle \mid \langle \text{control} \rangle$

$\langle \text{action} \rangle ::= \text{'AVANT'} \langle \text{expression-arithm} \rangle$   
|  $\text{'DROITE'} \langle \text{expression-arithm} \rangle$   
|  $\text{'GAUCHE'} \langle \text{expression-arithm} \rangle$   
|  $\text{'ALLERA'} \langle \text{expression-arithm} \rangle \text{' , ' } \langle \text{expression-arithm} \rangle$   
|  $\text{'POSER'}$   
|  $\text{'LEVER'}$   
|  $\text{'COULEUR'} \langle \text{couleur} \rangle$   
|  $\text{'EPAISSEUR'} \langle \text{expression-arithm} \rangle$

$\langle \text{couleur} \rangle ::= \text{'NOIR'}$

|  $\text{'BLANC'}$   
|  $\text{'GRIS'}$   
|  $\text{'BLEU'}$   
|  $\text{'VERT'}$   
|  $\text{'ROUGE'}$   
|  $\text{'JAUNE'}$   
|  $\text{'ROSE'}$   
|  $\text{'ORANGE'}$   
|  $\text{'VIOLET'}$   
|  $\text{'MARRON'}$

$\langle \text{entier} \rangle ::= ( \text{'0'} \mid \dots \mid \text{'9'} ) \{ \text{'0'} \mid \dots \mid \text{'9'} \}$

$\langle \text{control} \rangle ::= \text{'SI'} \langle \text{condition} \rangle \text{'ALORS'} \langle \text{script} \rangle \mid \text{'SINON'} \langle \text{script} \rangle \mid$   
|  $\text{'REPETER'} \langle \text{expression-arithm} \rangle \langle \text{script} \rangle$   
|  $\text{'TANTQUE'} \langle \text{condition} \rangle \langle \text{script} \rangle$

; La requête de contrôle 'REPETER' est une boucle "for"

$\langle \text{expression-arithm} \rangle ::= \langle \text{entier} \rangle$

$\langle \text{condition} \rangle ::= \langle \text{condition-simple} \rangle$   
|  $\text{'(' } \langle \text{expression-booleenne} \rangle \text{' )'}$

$\langle \text{expression-booleenne} \rangle ::= \langle \text{condition} \rangle$

$\langle \text{condition-simple} \rangle ::= \text{'ESTLEVE'}$   
|  $\langle \text{requete-position} \rangle \langle \text{op-compare} \rangle \langle \text{expression-arithm} \rangle$   
|  $\langle \text{requete-graphique} \rangle$

$\langle \text{requete-position} \rangle ::= \text{'POSX'} \mid \text{'POSY'}$

## 2 Fonctionnalités possibles

### 1) Les variables

Possibilités de déclarer des variables dans les scripts à n'importe quel endroit grâce à la directive " SOIT ( nom de variable )" suit d'un nombre ou expression arithmétique. Pour faire appel à une variable il faut utiliser " \$nomDeVariable"

Exemple de script :

```
SCRIPT
  SOIT T = 20
  AVANT $T
FIN
```

### 2 ) Courbes de Bézier

Pour utiliser les courbes de Bézier il faut utiliser la directive BEZIER suivit de deux points. La courbe commence à l'emplacement du crayon.

Exemple de script :

```
SCRIPT
  BEZIER 10 0 , 100 150
FIN
```

### 3 ) Les expressions booléennes

Les expressions booléennes peuvent être utilisées dans les nœuds " SI " et les nœuds "tant que " . Dans les conditions on peut utiliser les primitives POSX et POXY qui correspondent aux positions du crayon.

Exemples de scripts :

<pre>SCRIPT   SI POSX &gt; 200   ALORS SCRIPT     AVANT 20   FIN   SINON     AVANT 30   FIN FIN</pre>	<pre>SCRIPT   TANTQUE POSX &lt; 200   SCRIPT     AVANT 20   FIN FIN</pre>
---	---

#### 4 ) Les expressions arithmétique

Dans les scripts on peut utiliser des expressions arithmétique tel que " $10*(10+1)/3+(-2)$ ".

Exemple de script :

SCRIPT SI POSX > (2*10)/3 ALORS SCRIPT AVANT 20 FIN SINON AVANT 30 FIN FIN	SCRIPT SOIT V = 2*30 AVANT \$V FIN
--	---

#### 5 ) La gestion d'erreur

Les erreurs sont géré par des "pop-up" qui indique sur quelle instruction est l'erreur

Exemple :

