**Projet de compilation**

**Binôme : Sagno Félix & Cissé Mamadou**

**Fonctionnalités Disponible de notre projet de compilation**

Déclaration de variable,

Affectation des valeurs aux variables

Affichage des variables

Utilisation des conditions

Comparaison de variables dans une condition

Lecture d’une variable

Les commentaires sur une ligne

**Règles de prduction grammaticale :**

%start program

%%

program : listRetour

|

listDeclaration GUI INLINE listInstruction NEA listRetour

;

listDeclaration : declaration

|

listDeclaration declaration

;

declaration : IDENTIF INLINE;

listInstruction :instruction

|

listInstruction instruction

;

instruction : affichage

|

affectation

|

condition

|

lecture

|

commentaire

;

commentaire : COMMENT CHAINE INLINE

;

lecture : READ CRO\_O expression CRO\_F INLINE

;

affichage : WRITE CRO\_O contenu CRO\_F INLINE

;

contenu: simple\_affichage

|

texte

|

concat

;

concat: texte POINT simple\_affichage

|

simple\_affichage POINT texte

|

texte POINT simple\_affichage POINT texte

;

texte: GRIF list\_chaine GRIF

;

list\_chaine: CHAINE

|

list\_chaine CHAINE

;

simple\_affichage:ENTIER

|

IDENTIF

;

affectation : IDENTIF EGAL expression INLINE

;

condition: condition\_si POINT INLINE

|

condition\_si POINT INLINE condition\_sinon

|

condition\_si POINT INLINE list\_condition\_sinon\_si

;

list\_condition\_sinon\_si: condition\_sinon\_si POINT INLINE

|

condition\_sinon\_si POINT INLINE condition\_sinon

|

list\_condition\_sinon\_si condition\_sinon\_si POINT INLINE

;

condition\_sinon\_si:SINONSI CRO\_O comparaison CRO\_F INLINE listInstruction

;

condition\_si :SI CRO\_O comparaison CRO\_F INLINE listInstruction

;

condition\_sinon: SINON INLINE listInstruction POINT INLINE

;

comparaison: IDENTIF signe\_comparaison expression

;

signe\_comparaison: EGALCONDI

|

SUPEG

|

INFEG

|

INF

|

SUP

|

DIFF

;

expression : IDENTIF

|

ENTIER

;

listRetour: Retour

|

listRetour Retour

;

Retour : INLINE;

%%

**Exemple de programme source :**

@val1

@val2

Gui

    write["Debut du Programme"]

    @val=10

    @val2=09

    read[@val2]

    #[@val1==10]

        @val=20

    .

    //un commentaire ici

    write["fin du Programme"]

nea