

Grammaire du langage Z

Structure d'un algorithme

<Algo Z>	→	[~Soit Soient~ <Ps>] Debut <Lis> Fin [;] { ~<Act> <Fon>~ [;] }*
<Act>	→	Action Idf [()] [;] [~Soit Soient~ <Ps>] Debut <Lis> Fin
<Fon>	→	Fonction Idf () : <Typ> [~Soit Soient~ <Ps>] Debut <Lis> Fin
<Ps>	→	<S>;{ [~Soit Soient~] <S>; }*
<S>	→	[Sep ~<Typ> ~ Action Fonction (<Typ>)~ ~]
	→	Idf { , Idf }*

Les différents types

<Typ>	→	Types <Structsimple> <Structcomplexe> Machine_car Machine_nombre [Pointeur vers [Sep]] Liste [De ~Types <Structsimple>~ Tableau (<Lc>) [De ~<Structsimple> Types~] Fichier De ~ Types Vecteur (Cste) De ~Types <Structsimple> <Structcomplexe> ~ ~ Buffer [Entete] (Types { , types }*)
<Structsimple>	→	[Structure] (Types { , Types }*)
<Structcomplexe>	→	[Structure] (~ Types Vecteur (Cste) De Types ~ { , ~ Types Vecteur (Cste) De Types ~ }*)
<Lc>	→	Cste { , Cste }*

Les instructions

< Lis >	→	< Inst > { ; < Inst > }*
<Inst>	→	Idf := <Exp> Tantque <Exp> [:] <Lis> Fintantque Si <Exp> [:] <Lis> [Sinon <Lis>] Fsi Pour Idf := <Exp> , <Exp> [, <Exp>] [:] <Lis> Finpour Appel Idf [(Exp { , <Exp> }*)] Lire (Idf { , Idf }*) Ecrire (<Exp> { , <Exp> }*) ~ Liberer Allouer Fermer ~ (<Exp>) Ouvrir ((Idf , Chaine , Chaine) ~ Lirecar Lirenomb Lireseq Ecrireseq Rajouter ~ (Idf , Idf)

~ **Aff_adr**|**Aff_val**~ (<Exp> , <Exp>) |
 ~ **Creer_liste** |**Init_vecteur**|**Creer_mnombre**|**Init_struct**~
 (Idf , [[~<Exp>|[[<Exp> {, <Exp>}*]] ~
 {, ~<Exp>|[[<Exp> {, <Exp>}*]] ~ }*]]) |
Creer_mcar(Idf, [[Chaine]]) |
Aff_element (<Exp> [[<Exp> {, <Exp> }*]] , <Exp>) |
 ~**Aff_struct**|**Aff_entete**~(Idf, Cste, <Exp>) |
 ~ **Liredir** | **Ecredir** ~ (Idf, Cste, Idf)

Les expressions

<Exp> → <Exps>[Opr <Exps>]
 <Exps> → [Sign] <Terme> { Opa <Terme> }* |
 <Terme> → <Facteur>{Opm <Facteur>}*
 <Facteur> → **Idf** [(<Exp> {, <Exp>} *)] | **Cste** |
 <Fonct> | **NIL** | (<Exp>) |
 Chaine | **Non** <Facteur> | Cstelogic
 <Fonct> → ~**Valeur**|**Suivant**~ (<Fonct>) |
Element (<Fonct> [[<Exp> {, <Exp>}*]])
 ~**Struct**|**Entete**|**Caract**~ (Idf, Cste) |
 ~ **Aleachaine** | **Aleanombre** | **Longchaine**~ (<Exp>) |
 ~**Nbrcar**|**NbrNombre**|**Finfich**|**Alloc_bloc**~ (Idf)