Main\_Program ::= Main\_Program’’ Main\_Program’ ✅

Main\_Program’ ::= Main\_Program Main\_Program’ | ε ✅

Main\_Program’’ ::= Declaration | Assignment | City | For ✅

Declaration ::= **const** Declaration’ | Declaration’ ✅

Declaration’ ::= Type **variable equals** Expression ✅

Assignment ::= **variable equals** Expression ✅

Type ::= **type\_number** | **type\_string** | **type\_coordinate** | **type\_list lthen** Type **mthen**✅

Expression ::= Additive ✅

Additive ::= Multiplicative Additive’ ✅

Additive’ ::= **plus** Multiplicative Additive’ | **minus** Multiplicative Additive’ | ε ✅

Multiplicative ::= Exponential Multiplicative’ ✅

Multiplicative’ ::= **times** Exponential Multiplicative’ | **divide** Exponential Multiplicative’ | **integer-divide** Exponential Multiplicative’ | ε ✅

Exponential ::= Unary Exponential’ ✅

Exponential’ ::= **pow** Unary Exponential’ | ε ✅

Unary ::= **plus** Primary | **minus** Primary | Primary ✅

Primary ::= **real** | **variable** | **lbracket** Additive **rbracket** | **lsq\_bracket** Inner\_List **rsq\_bracket** ✅

Inner\_List ::= Inner\_List’’ Inner\_List’ ✅

Inner\_List’ ::= **comma** Inner\_List’’ Inner\_List | ε ✅

Inner\_List’’ ::= Expression✅

City ::= **city variable block\_start** City’ **block\_end** ✅

City’ ::= City’’’ City’’ ✅

City’’ ::= City’ City’’ | ε ✅

City’’’ ::= City\_Constructs | Declaration | Assignment | Print ✅

Print ::= **print lbracket** Expression **rbracket** ✅

City\_Constructs ::= Restaurant | Road ✅

Restaurant ::= Name Shape Marker Routes ✅

Name ::= **name colon** String ✅

String ::= **variable | string** ✅

Shape ::= **shape colon block\_start** Lines **block\_end** ✅

Lines ::= Line Line Line Line’ ✅

Line’ ::= Line Line’ | ε ✅

Line ::= **line lsq\_bracket** Coord **comma** Coord **rsq\_bracket** ✅

Coord ::= **variable** | **lbracket** Expression **comma** Expression **rbracket** ✅

Marker ::= Marker’ | ε ✅

Marker’ ::= **marker colon** Point ✅

Point ::= **point lsq\_bracket** Coord **rsq\_bracket** ✅

Routes ::= Routes’ | ε ✅

Routes’ ::= **routes colon lsq\_bracket** Roads **rsq\_bracket** ✅

Roads ::= Road Road’ ✅

Road’ ::= **comma** Road Roads’ | ε ✅

Road ::= **road variable block\_start** NameRoad\_Shapes **block\_end** ✅

Road\_Shapes ::= **shape colon block\_start** Road\_Shapes’ Road\_Shapes **block\_end**✅

Road\_Shapes’ ::= Road\_Shape Road\_Shapes’ | ε ✅

Road\_Shape ::= Line | Bend ✅

Bend ::= **bend lsq\_bracket** Coord **comma** Coord **comma** Expression **rsq\_bracket** ✅

For ::= **foreach variable in** Variable **block\_start** Program **block\_end**✅

Variable ::= **variable** | Radius ✅

Radius ::= **lsq\_bracket** Coord **comma** Expression **rsq\_bracket** ✅

Program ::= Program’’ Program’ ✅

Program ‘::= Program Program‘ | ε ✅

Program ‘’::= Declaration | Assignment | Print | Highlight **hash** ✅

Highlight ::= **highlight lbracket variable rbracket** ✅pazi da je variable coord!

**const ->** const

**variable ->** {A,…,Z,a,…,z}{A,…,Z,a,…,z,0,…,9}\*

**equals ->** =

**type\_number ->** num

**type\_string ->** string

**type\_coordinate ->** coord

**type\_list ->** List

**lthen ->** <

**mthen ->** >

**city ->** city

**block\_start ->** {

**block\_end ->** }

**name ->** name

**colon ->** :

**string ->** “{A,…,Z,a,…,z,0,…,9}\*” oz. ASCII

**shape ->** shape

**line ->** line

**lsq\_bracket ->** [

**comma** **->** ,

**rsq\_bracket ->** ]

**lbracket ->** (

**rbracket ->** )

**marker  ->** marker

**point->** point

**routes->** routes

**road->** road

**bend->** bend

**foreach->** foreach

**highlight ->** highlight

**hash ->** #