

名称	説明
Agt	エージェントそのもの。エージェントの実体値
AgtSet	エージェントの集合
Boolean	真のときはTrue、偽のときはFalse
Double	以下の範囲の小数点以下が何桁もある数値(擬似的な実数)(負の場合) -1.79769313486232*10308 to -4.94065645841247*10 ⁻³²⁴ (正の場合) 4.94065645841247*10324 to 1.79769313486232*10308
Integer	以下の範囲の整数 -2,147,483,648 to 2,147,483,647
Long	以下の範囲の整数 -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
Space	モデルツリーで定義される空間 ※空間のサイズは縦幅 1 to 10,000、横幅 1 to 10,000
String	文字数は 0 to (無制限)

Agt

エージェントそのもの。エージェントの実体値

[形式]

エージェント型
Dim one As Agt (oneはエージェント型変数)

AgtSet

エージェントの集合

[形式]

エージェント集合型

Boolean

真のときはTrue、偽のときはFalse

[形式]

ブール型
変数 = True または False

Double

以下の範囲の小数点以下が何桁もある数値(擬似的な実数)(負の場合) -1.79769313486232*10308 to
-4.94065645841247*10⁻³²⁴ (正の場合) 4.94065645841247*10324 to 1.79769313486232*10308

[形式]

実数型
Dim one As Double (oneは実数型変数)

Integer

以下の範囲の整数 -2,147,483,648 to 2,147,483,647

[形式]

整数型

Dim one As Integer (oneは整数型変数)

Long

以下の範囲の整数 -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807

[形式]

長整数型

Dim one As Long (oneは長整数型変数)

Space

モデルツリーで定義される空間 ※空間のサイズは縦幅 1 to 10,000、横幅 1 to 10,000

[形式]

空間型

String

文字数は 0 to (無制限)

[形式]

文字列型

Dim one As String (oneは文字列型変数)

算術演算子

名称	説明
*	乗算
+	加算
-	減算
/	商を実数値として求める除算
Mod	整数の除算の余り(剰余) (小数点以下の値は切り捨て)
¥	商を整数値として求める除算 (小数点以下の値は切り捨て)
^	べき乗

*

乗算

[形式]

one = 2 * 3 (oneに 6を代入)

+

加算

[形式]

one = 1 + 2 (oneに 3を代入)

-

減算

[形式]

one = 2 - 1 (oneに -1を代入)

/

商を実数値として求める除算

[形式]

one = 5 / 3 (oneに 1.6666... を代入)
two = 5 / 3.0 (twoに 1.6666... を代入)

Mod

整数の除算の余り(剰余) (小数点以下の値は切り捨て)

[形式]

one = 5 Mod 3 (oneに 2を代入)
two = 7 Mod 3 (twoに 1を代入)

¥

商を整数値として求める除算 (少数点以下の値は切り捨て)

[形式]

one = 5 ¥ 3 (oneに 1を代入)
two = 5 ¥ 3.0 (two に 1を代入)

^

べき乗

[形式]

one = 2 ^ 3 (oneに 8を代入)
two = 2 ^ 4 (twoに 16を代入)

文字列演算子

名称	説明
&	文字列の連結

&

文字列の連結

[形式]

one = "a" & "b" (oneに abを代入)

関係演算子

名称	説明
!=	左辺と右辺が等しくない
==	左辺と右辺が等しい ※VisualBasicでは、「=」であるため注意
>	左辺が右辺より大きい
>=	左辺が右辺より大きいか等しい(以上)
<	左辺が右辺より小さい
<=	左辺が右辺より小さいか等しい(以下)
<>	左辺と右辺が等しくない

!=

左辺と右辺が等しくない

[形式]

one != 3 (oneは 3と等しくない)

==

左辺と右辺が等しい ※VisualBasicでは、「=」であるため注意

[形式]

one == 3 (oneは 3と等しい)

>

左辺が右辺より大きい

[形式]

one > 3 (oneは 3より大きい)

>=

左辺が右辺より大きいか等しい(以上)

[形式]

one >= 3 (oneは 3以上)

<

左辺が右辺より小さい

[形式]

$one < 3$ (oneは 3より小さい)

\leq

左辺が右辺より小さいか等しい(以下)

[形式]

$one \leq 3$ (oneは 3以下)

\diamond

左辺と右辺が等しくない

[形式]

$one \diamond 3$ (oneは 3と等しくない)

論理演算子

名称	説明
And	かつ(論理積)
Not	ではない(否定)
Or	または(論理和)
Xor	どちらかが真(排他的論理和)

And

かつ(論理積)

[形式]

A And B (AかつB)

Not

ではない(否定)

[形式]

Or

または(論理和)

[形式]

A Or B (AまたはB)

Xor

どちらかが真(排他的論理和)

[形式]

A Xor B (AまたはBのどちらかが真)

代入演算子

名称	説明
=	式の結果を代入する(代入)

=

式の結果を代入する(代入)

[形式]

one = 1 + 2 (oneに 3を代入)
two = 2 * 3 (twoに 6を代入)

条件判断文

名称	説明
IF文	条件によって処理を分岐させる

IF文

条件によって処理を分岐させる

[形式]

- (1) 式の値が真の場合にブロック文を実行する
If 式 Then
 ブロック文
End If
- (2) 式の値が真の場合はブロック文1、偽の場合はブロック文2を実行する
If 式 Then
 ブロック文1
Else
 ブロック文2
End If
- (3) 式の値が真の場合に、対応するブロック文を実行する
If 式1 Then
 ブロック文1
Elseif 式2 Then
 ブロック文2
Elseif 式3 Then
 ブロック文3
Else
 上記条件以外の場合
EndIf

名称	説明
Do Until 文	条件式が満たされるまで(式の値が偽の間)繰り返す
Do While 文	条件式が満たされている間(式の値が真の間)繰り返す
For Each ? In ? Next 文	エージェント集合の各エージェントに対して同じ操作を行う
For ? To ? Next 文	処理を所定回数繰り返す

Do Until 文

条件式が満たされるまで(式の値が偽の間)繰り返す

[形式]

Do Until 式
 ブロック文
Loop

Do While 文

条件式が満たされている間(式の値が真の間)繰り返す

[形式]

Do While 式
 ブロック文
Loop

For Each ? In ? Next 文

エージェント集合の各エージェントに対して同じ操作を行う

[形式]

For Each ループエージェント In エージェント集合
 ブロック文
Next ループエージェント

For ? To ? Next 文

処理を所定回数繰り返す

[形式]

For ループ変数 = 初期値 To 最終値 (Step きざみ値、省略時は1) ブロック文
Next ループ変数

※Step きざみ値は正の整数を指定してください。

名称	説明
Break 文	繰り返し文による繰り返し作業を中止する

Break 文

繰り返し文による繰り返し作業を中止する

[形式]

(1) While文のループを途中で抜ける場合

```
Do While 条件式
  Do While 条件式
    If 脱出条件式 Then
      Break      ⇒※1つ外に脱出する
    End If
  Loop
Loop
```

(2) For文のループを途中で抜ける場合

```
For i = 初期値 To 最終値
  For j = 初期値 To 最終値
    If 脱出条件式 Then
      Break      ⇒一つ外に脱出する
    End If
  Next j
Next i
```

名称	説明
Include 文	外部ファイル(以下、インクルードファイルとする)に定義した関数を読み込む。よく利用する関数をインクルードファイルに記述することで、複数のエージェントから同一の関数を呼ぶことができる。通常はルールエディタの先頭行に記述する

Include 文

外部ファイル(以下、インクルードファイルとする)に定義した関数を読み込む。よく利用する関数をインクルードファイルに記述することで、複数のエージェントから同一の関数を呼ぶことができる。通常はルールエディタの先頭行に記述する

[形式]

include “インクルードファイル名”

- ※インクルードファイルの作成: テキストエディタに自作のユーザ定義関数をコピーし、拡張子を「.inc」として保存する
- ※インクルードファイルの使用環境整備: インクルードファイルをモデルファイルと同じフォルダに保存する
- ※インクルードファイル中の関数の使用: ルールエディタの先頭に、「include “インクルードファイル名”」と記述した後は、組み込み関数と同様に使用する
- ※include文は、関数内でいくつ記述しても構わない
- ※include文を複数定義する場合は、1ファイル1行で記述する
- ※Universeで定義した関数およびUniverseでインクルードファイルとして定義された関数は、エージェントから「@関数名」で呼び出せる

コメントアウト

名称	説明
<code>/** */</code>	「/*」から「*/」までの区間がコメントとなり、ルールに反映されなくなる
<code>//</code>	「//」から行の終わりまでがコメントとなり、ルールに反映されなくなる

`/** */`

「/*」から「*/」までの区間がコメントとなり、ルールに反映されなくなる

[形式]

```
/*  
  コメント  
*/
```

`//`

「//」から行の終わりまでがコメントとなり、ルールに反映されなくなる

[形式]

```
// コメント
```

その他

名称	説明
Return 文	ユーザ定義関数の実行を終了して、呼び出し側に実行の制御を戻す。関数が戻り値を返す関数として定義してあれば、値を返す

Return 文

ユーザ定義関数の実行を終了して、呼び出し側に実行の制御を戻す。関数が戻り値を返す関数として定義してあれば、値を返す

[形式]

Return (式)

※Return文は、関数内でいくつ記述しても構わない

※以下の場合には翻訳エラーとなる

- ・ 値を返す関数として定義し、式を省略
- ・ 値を返す関数として定義し、戻り値の型と式が異なる場合
- ・ 値を返さない関数として定義し、式を記述した場合

数値計算

名称	説明
Abs	絶対値を求める
Atn	アークタンジェントの値を求める
Cos	コサインの値を求める
Cosh	ハイパボリックコサインの値を求める
DegreeToRad	角度の単位を度からラジアンに変換する
Exp	自然対数eを底とする数値のべき乗を返す
FlatDegree	角度 (degree)を 0° to 360° の値に変換する
FlatRad	角度 (Rad)を 0 to 2π の値に変換する
GetRandomSeed	乱数シード値を取得する
Int	実数値の少数部分を切り捨てて、整数化する
Log	自然対数eを底としたln(x)の値を返す
NormDist	指定した平均と標準偏差に対する正規分布関数の値を返す
NormInv	指定した平均と標準偏差に対する正規累積分布関数の逆関数の値を返す(正規分布に従うような値をランダムに求める際に使用する)
PI	πの値を取得する
PoissonRnd	ポアソン分布で与えられる乱数を取得する
RadToDegree	角度(rad)を角度(degree)へ変換する
Rnd	0.0以上 1.0未満の一樣乱数値(ランダムな値)を求める
Round	小数点以下を四捨五入する
SetRandomSeed	乱数シード値を設定する
Sin	サインの値を求める
Sinh	ハイパボリックサインの値を求める
Sqr	平方根を求める
Tan	タンジェントの値を求める
Tanh	ハイパボリックタンジェントの値を求める

Abs

絶対値を求める

[形式]

Abs (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値

[戻り値]

型	説明
Double	絶対値

Atn

アークタンジェントの値を求める

[形式]

Atn(arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値:単位はラジアン

[戻り値]

型	説明
Double	アークタンジェントの値: $-\pi/2? \pi/2$

Cos

コサインの値を求める

[形式]

Cos(arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値:単位はラジアン

[戻り値]

型	説明
Double	コサインの値

Cosh

ハイパボリックコサインの値を求める

[形式]

Cosh(arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値:単位はラジアン

[戻り値]

型	説明
Double	ハイパボリックコサインの値

DegreeToRad

角度の単位を度からラジアンに変換する

[形式]

DegreeToRad(arg1)

[引数]

引数名	型	説明
arg1	Double	角度(degree)

[戻り値]

型	説明
Double	角度 (Rad)

Exp

自然対数eを底とする数値のべき乗を返す

[形式]

Exp(arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値

[戻り値]

型	説明
Double	自然対数eを底とする数値のべき乗

FlatDegree

角度 (degree)を 0° to 360° の値に変換する

[形式]

FlatDegree(arg1)

[引数]

引数名	型	説明
arg1	Double	角度(degree)

[戻り値]

型	説明
Double	角度 (degree)

FlatRad

角度 (Rad)を 0 to 2π の値に変換する

[形式]

FlatRad(arg1)

[引数]

引数名	型	説明
arg1	Double	角度(rad)

[戻り値]

型	説明
Double	角度 (rad)

GetRandomSeed

乱数シード値を取得する

[形式]

GetRandomSeed ()

[戻り値]

型	説明
Long	乱数シード値

Int

実数値の少数部分を切り捨てて、整数化する

[形式]

Int(arg1)

(記述例) この場合の戻り値は3
Int(3.1415)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値

[戻り値]

型	説明
Integer	正数値

Log

自然対数eを底としたln(x)の値を返す

[形式]

Log(arg1)

[引数]

引数名	型	説明
-----	---	----

arg1	Double, Integer, Long	正または負の値
------	-----------------------	---------

[戻り値]

型	説明
Double	ln(x)の値

NormDist

指定した平均と標準偏差に対する正規分布関数の値を返す

[形式]

NormDist(arg1, arg2, arg3, arg4)

[引数]

引数名	型	説明
arg1	Double	関数に代入する値
arg2	Double	対象となる分布の算術平均(相加平均)
arg3	Double	対象となる分布の標準偏差
arg4	Boolean	関数形式 ※Trueを指定すると累積分布関数の値を計算 ※Falseを指定すると確率密度関数の値を計算

[戻り値]

型	説明
Double	指定した平均と標準偏差に対する正規分布関数の値

NormInv

指定した平均と標準偏差に対する正規累積分布関数の逆関数の値を返す(正規分布に従うような値をランダムに求める際に使用する)

[形式]

NormInv(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Double	正規分布における確率(0超過 1未満の一樣乱数)
arg2	Double	対象となる分布の算術平均(相加平均)
arg3	Double	対象となる分布の標準偏差

[戻り値]

型	説明
Double	正規累積分布関数の逆関数の値

PI

πの値を取得する

[形式]

PI()

[戻り値]

型	説明
Double	π の値

PoissonRnd

ポアソン分布で与えられる乱数を取得する

[形式]

PoissonRnd (arg1)

[引数]

引数名	型	説明
arg1	Double	λ の値 (平均値)、0より大きい実数値

[戻り値]

型	説明
Integer	ポアソン分布に従って得られた乱数値

RadToDegree

角度(rad)を角度(degree)へ変換する

[形式]

RadToDegree (arg1)

[引数]

引数名	型	説明
arg1	Double	角度(Rad)

[戻り値]

型	説明
Double	角度(degree)

Rnd

0.0以上 1.0未満の一樣乱数値(ランダムな値)を求める

[形式]

Rnd ()

[戻り値]

型	説明
Double	乱数値

Round

小数点以下を四捨五入する

[形式]

Round (arg1)

[引数]

引数名	型	説明
arg1	Double	正または負の値

[戻り値]

型	説明
Integer	整数値

SetRandomSeed

乱数シード値を設定する

[形式]

SetRandomSeed (arg1)

[引数]

引数名	型	説明
arg1	Long	乱数シード値

Sin

サインの値を求める

[形式]

Sin (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値:単位はラジアン

[戻り値]

型	説明
Double	サインの値

Sinh

ハイパボリックサインの値を求める

[形式]

Sinh (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値:単位はラジアン

[戻り値]

型	説明
Double	ハイパボリックサインの値

Sqr

平方根を求める

[形式]

Sqr (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値

[戻り値]

型	説明
Double	平方根の値

Tan

タンジェントの値を求める

[形式]

Tan (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値:単位はラジアン

[戻り値]

型	説明
Double	タンジェントの値

Tanh

ハイパボリックタンジェントの値を求める

[形式]

Tanh (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long	正または負の値:単位はラジアン

[戻り値]

型	説明
Double	ハイパボリックタンジェントの値

名称	説明
CountToken	カンマ区切りの文字列の部分文字列数を取得する。カンマの数から判定し、文字列が空の場合は0、文字列は空ではないがカンマがない場合は1とする。
GetToken	カンマ区切りの文字列から指定した位置の文字列を取得する
InStr	対象文字列の中から指定した比較文字列を検索し、最初に見つかった文字位置(対象文字列の何文字目)を返す
Left	文字列の左端から、指定文字数分の文字列を抽出する
Len	文字列の文字数を求める
Mid	文字列の指定位置から指定文字数分の文字列を抽出する
Replace	文字列を置換する
Right	文字列の右端から、指定文字数分の文字列を抽出する
StrComp	文字列を比較する
Trim	文字列の前後の半角空白文字を削除する

CountToken

カンマ区切りの文字列の部分文字列数を取得する。カンマの数から判定し、文字列が空の場合は0、文字列は空ではないがカンマがない場合は1とする。

[形式]

CountToken (arg1)

[引数]

引数名	型	説明
arg1	String	文字列

[戻り値]

型	説明
Integer	部分文字列の数

GetToken

カンマ区切りの文字列から指定した位置の文字列を取得する

[形式]

GetToken (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	文字列
arg2	Integer	取得したい文字列の位置(0以上の整数)

[戻り値]

型	説明
String	取得した文字列(失敗時はnullを返す)

InStr

対象文字列の中から指定した比較文字列を検索し、最初に見つかった文字位置(対象文字列の何文字目)を返す

[形式]

InStr(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Integer	検索の開始位置(1以上)
arg2	String	対象文字列
arg3	String	比較文字列

[戻り値]

型	説明
Integer	文字位置(見つからなかった時は0を返す)

Left

文字列の左端から、指定文字数分の文字列を抽出する

[形式]

Left(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	対象文字列
arg2	Integer	抽出文字数

[戻り値]

型	説明
String	抽出した文字列

Len

文字列の文字数を求める

[形式]

Len(arg1)

[引数]

引数名	型	説明
arg1	String	対象文字列

[戻り値]

型	説明
Integer	文字列の文字数

Mid

文字列の指定位置から指定文字数分の文字列を抽出する

[形式]

Mid(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	String	対象文字列
arg2	Integer	抽出開始位置(1以上)
arg3	Integer	抽出文字数

[戻り値]

型	説明
String	抽出した文字列

Replace

文字列を置換する

[形式]

Replace(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	String	対象文字列
arg2	String	検索する文字列
arg3	String	置換する文字列

[戻り値]

型	説明
String	置換後の文字列

Right

文字列の右端から、指定文字数分の文字列を抽出する

[形式]

Right(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	対象文字列
arg2	Integer	抽出文字数

[戻り値]

型	説明
---	----

String	抽出した文字列
--------	---------

StrComp

文字列を比較する

[形式]

StrComp(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	対象文字列
arg2	String	比較文字列

[戻り値]

型	説明
Integer	判定値（※判定値が =0のとき、arg1とarg2は完全一致 <0のとき、arg1が昇順ソートの上位側、>0のとき、arg1が昇順ソートの下位側）

Trim

文字列の前後の半角空白文字を削除する

[形式]

Trim(arg1)

[引数]

引数名	型	説明
arg1	String	対象文字列

[戻り値]

型	説明
String	前後の半角空白文字を削除した文字列

名称	説明
CAgt	整数型(Integer)、文字列型(String)の値をエージェント型(Agt)に変換する
CAgtType	文字列型(String)の値をエージェント種別型(AgtType)に変換する
CBool	実数型(Double)、整数型(Integer)、長整数型(Long)、文字列型(String)の値が0のときはFalse、それ以外の場合はTrueに変換する
CDbl	ブール型(Boolean)、整数型(Integer)、長整数型(Long)、文字列型(String)の値を実数型(Double)に変換する
CInt	ブール型(Boolean)、実数型(Double)、文字列型(String)、エージェント型(Agt)の値を整数型(Integer)に変換する
CLong	ブール型(Boolean)、実数型(Double)、整数型(Integer)、文字列型(String)の値を長整数型(Long)に変換する
CSpace	文字列型(String)の値を空間型(Space)に変換する
CStr	ブール型(Boolean)、実数型(Double)、整数型(Integer)、長整数型(Long)、空間型(Space)、エージェント種別型(AgtType)、エージェント型(Agt)の値を文字列型(String)に変換する

CAgt

整数型(Integer)、文字列型(String)の値をエージェント型(Agt)に変換する

[形式]

CAgt(arg1)

[引数]

引数名	型	説明
arg1	Integer, String	変換前の値

[戻り値]

型	説明
Agt	変換後の値

CAgtType

文字列型(String)の値をエージェント種別型(AgtType)に変換する

[形式]

CAgtType(arg1)

[引数]

引数名	型	説明
arg1	String	変換前の値

[戻り値]

型	説明
AgtType	変換後の値

CBool

実数型(Double)、整数型(Integer)、長整数型(Long)、文字列型(String)の値が0のときはFalse、それ以外の場合はTrueに変換する

[形式]

CBool (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long, String	変換前の値

[戻り値]

型	説明
Boolean	変換後の値

CDbl

ブール型(Boolean)、整数型(Integer)、長整数型(Long)、文字列型(String)の値を実数型(Double)に変換する

[形式]

CDbl (arg1)

[引数]

引数名	型	説明
arg1	Boolean, Integer, Long, String	変換前の値

[戻り値]

型	説明
Double	変換後の値

CInt

ブール型(Boolean)、実数型(Double)、文字列型(String)、エージェント型(Agt)の値を整数型(Integer)に変換する

[形式]

CInt (arg1)

[引数]

引数名	型	説明
arg1	Boolean, Double, String, Agt	変換前の値

[戻り値]

型	説明
Integer	変換後の値

CLong

ブール型(Boolean)、実数型(Double)、整数型(Integer)、文字列型(String)の値を長整数型(Long)に変換する

[形式]

CLong (arg1)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, String,	変換前の値

[戻り値]

型	説明
Long	変換後の値

CSpace

文字列型(String)の値を空間型(Space)に変換する

[形式]

CSpace (arg1)

[引数]

引数名	型	説明
arg1	String	変換前の値

[戻り値]

型	説明
Space	変換後の値

CStr

ブール型(Boolean)、実数型(Double)、整数型(Integer)、長整数型(Long)、空間型(Space)、エージェント種別型(AgtType)、エージェント型(Agt)の値を文字列型(String)に変換する

[形式]

CStr (arg1)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, Long, Space, AgtType, Agt	変換前の値

[戻り値]

型	説明
String	変換後の値

エージェント操作

名称	説明
AddAgt	エージェント集合型変数にエージェントを追加する
ClearAgtSet	エージェント集合型変数の中身をクリアする
CopyAgtSet	エージェント集合型変数1をエージェント集合型変数2にコピーする。※可能な限り、CopyAgtSetではなく、DuplicateAgtSetを利用する
CountAgt	指定されたエージェント種別のエージェント数を取得する
CountAgtSet	エージェント集合型変数が保持しているエージェント数を取得する
CountAliveAgt	指定されたエージェント種別のうち、Killフラグの立っていないエージェント数を取得する ※KillAgtにより、エージェントが実際に削除されるのは、ステップの終了時である
CreateAgt	エージェントを1つ生成する
DelAgt	エージェントを削除する ※エージェントはルール実行中であっても即座に削除される
DelAgtSet	エージェント集合型変数1から、エージェント集合型変数2に含まれるエージェントを削除する
DelAgtSet2	エージェント集合型変数1から、指定したエージェントを削除する
DuplicateAgtSet	エージェント集合型変数2を、エージェント集合型変数1にコピーする
Forward	カッコ内の数だけそのエージェントが向いている方向へ進む(進行方向はDirection変数による)。戻り値は、正常終了のときは-1、空間がループしていないときで、指定した距離を進めなかった時は、進めなかった距離を返す
ForwardDirectionCell	セル上を指定した方向へ移動する
ForwardX	カッコ内の数だけX軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す
ForwardXCell	セル上をカッコ内の数だけX軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す
ForwardY	カッコ内の数だけY軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す
ForwardYCell	セル上をカッコ内の数だけY軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す
GetAgt	エージェント集合型変数の指定位置(0以上の整数)にあるエージェントを取得する
GetAgtEntry	エージェント集合型変数に指定エージェントが含まれているかを調査する。指定エージェントが含まれている場合、エージェント集合型変数の先頭からの連番(要素番号)のうち最も小さい値を0以上の整数で返す。含まれていなかった場合は-1を返す
GetDirection	A地点からB地点への角度を求める。空間がループする場合は、最短距離として求められる方向の角度を求める
GetHeightSpaceOwn	エージェント自身が乗っている空間の縦幅を取得する
GetHistory	記憶した変数値を取得する
GetUniqueID	エージェントIDからUniqueIDを取得する

GetWidthSpaceOwn	エージェント自身が乗っている空間の横幅を取得する
JoinAgtSet	エージェント集合型変数1にエージェント集合型変数2を追加する(要素の重複を認める)
KillAgt	エージェントを削除する(エージェントが実際に削除されるのは、ステップの終了時)
MakeAgtSet	指定したエージェント種別のエージェント全てを、エージェント集合型変数に格納する
MakeAgtSetAroundOwnCell	指定したエージェント集合の中から、エージェント自身の周り(セル上)にいるエージェントを選び、エージェント集合型変数に格納する
MakeAgtSetAroundPositionCell	指定したエージェント集合の中から、指定した位置座標の周り(セル上)にいるエージェントを選び、指定したエージェント集合型変数内に格納する
MakeAgtSetSpace	指定した空間上に存在する全てのエージェントを指定したエージェント集合型変数に格納する
MakeAllAgtSetAroundOwn	エージェント自身の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野の範囲の周囲にいる全てのエージェントをリストアップ)
MakeAllAgtSetAroundOwnCell	エージェント自身の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野(格子)の範囲の周囲にいる全てのエージェントをリストアップ)
MakeAllAgtSetAroundPosition	指定した位置座標の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、指定した位置座標の周囲にいる全てのエージェントをリストアップ)
MakeAllAgtSetAroundPositionCell	セル上に指定した位置座標の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、セル上に指定した位置座標の周囲にいる全てのエージェントをリストアップ)
MakeCommonAgtSet	エージェント集合型変数2とエージェント集合型変数3のどちらにも含まれているエージェントからなるエージェント集合型変数1を取得する
MakeDiffAgtSet	エージェント集合型変数2とエージェント集合型変数3のどちらか一方にだけ含まれているエージェントからなるエージェント集合型変数1を取得する
MakeOneAgtSetAroundOwn	エージェント自身の周り(セル上)にいる指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野の範囲の周囲にいる指定種別のエージェントをリストアップ)
MakeOneAgtSetAroundOwnCell	エージェント自身の周り(セル上)にいる指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野(格子)の範囲の周囲にいる指定種別のエージェントをリストアップ)
MakeOneAgtSetAroundPosition	指定した位置座標の周り(セル上)にいる指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、指定した位置座標の周囲にいる指定種別のエージェントをリストアップ)
MakeOneAgtSetAroundPositionCell	セル上に指定した位置座標の周り(セル上)にいる指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、セル上に指定した位置座標の周囲にいる指定種別のエージェントをリストアップ)
MergeAgtSet	エージェント集合型変数1にエージェント集合型変数2を追加する(要素の重複は認めない)
MoveLayerSpace	移動させたいLayer数だけLayerを移動させる

MoveToCenter	エージェント自身が乗っている空間(Layer)の中央に移動する
MoveToSpaceAgtSetCell	指定された座標を中心に、指定エージェント集合型変数に属するエージェントには重ならないように周りの空き地(セル)を探し、移動する。C
MoveToSpaceOwnCell	エージェント自身の周りの空き地(セル)を探し、移動する。(検索された領域に複数の空き地が存在するときは、ランダムに移動位置を決定)
MoveToSpacePositionCell	指定された座標を中心に、周りの空き地(セル)を探し、移動する。(検索された領域に複数の空き地が存在するときは、ランダムに移動位置を決定)
OnLayerSpace	指定したLayerへ移動する
PickupAgt	エージェント集合型変数の指定位置(0以上の整数)にあるエージェントを取得し、取得したエージェントをエージェント集合型変数から削除する
PurifyAgtSet	エージェント集合型変数2の中からエージェントの重複を取り除いたエージェント集合型変数1を取得する
Pursue	目標としたエージェントに向かって移動する。移動方向に従い、Direction変数の値が更新される
RandomPutAgtSet	指定されたエージェント集合をランダムに配置する
RandomPutAgtSetCell	指定されたエージェント集合をセル上にランダムに配置する
RemoveAgt	エージェント集合型変数から指定エージェントを削除する
ReverseDirectionCell	セル上で逆方向を取得する
SortAgtSet	エージェント集合型変数を指定された変数をキーとしてソートする
SpecifyAgtType	指定したエージェントのエージェント種別を取得する
SpecifyKillAgt	Killフラグの値を取得する
TerminateAgt	エージェントを完全削除する ※エージェントはルール実行中であっても即座に削除される ※エージェント型変数のうち、該当エージェントが含まれる変数は-1が代入される ※エージェント集合型変数に該当エージェントが含まれる場合は、エージェント集合型変数からも削除される
Turn	向きを変える(Direction変数に反映)。引数が正の値のときは左回り、負の値のときは右回りに向きを変える(左下原点の場合)。※この関数によりDirection変数は0?360の値を取る ※左上原点の設定時は回転方向が逆になる。
TurnAgt	指定されたエージェントの方向を向く

AddAgt

エージェント集合型変数にエージェントを追加する

[形式]

AddAgt(arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Agt	エージェント型変数

ClearAgtSet

エージェント集合型変数の中身をクリアする

[形式]

ClearAgtSet (arg1)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数

CopyAgtSet

エージェント集合型変数1をエージェント集合型変数2にコピーする。※可能な限り、CopyAgtSetではなく、DuplicateAgtSetを利用する

[形式]

CopyAgtSet (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2

CountAgt

指定されたエージェント種別のエージェント数を取得する

[形式]

CountAgt (arg1)

[引数]

引数名	型	説明
arg1	AgtType	エージェント種別型変数

[戻り値]

型	説明
Integer	エージェント数

CountAgtSet

エージェント集合型変数が保持しているエージェント数を取得する

[形式]

CountAgtSet (arg1)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数

[戻り値]

型	説明
Integer	エージェント数

CountAliveAgt

指定されたエージェント種別のうち、Killフラグの立っていないエージェント数を取得する ※KillAgtにより、エージェントが実際に削除されるのは、ステップの終了時である

[形式]

CountAliveAgt (arg1)

[引数]

引数名	型	説明
arg1	AgtType	エージェント種別型変数

[戻り値]

型	説明
Integer	Killフラグの立っていないエージェント数

CreateAgt

エージェントを1つ生成する

[形式]

CreateAgt (arg1)

[引数]

引数名	型	説明
arg1	AgtType	エージェント種別型変数

[戻り値]

型	説明
Agt	エージェント型変数

DelAgt

エージェントを削除する ※エージェントはルール実行中であっても即座に削除される

[形式]

DelAgt (arg1)

[引数]

引数名	型	説明
arg1	Agt	エージェント型変数

[戻り値]

型	説明
Boolean	判定値（※成功のときはTrue、失敗のときはFalse）

DelAgtSet

エージェント集合型変数1から、エージェント集合型変数2に含まれるエージェントを削除する

[形式]

DelAgtSet (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2

DelAgtSet2

エージェント集合型変数1から、指定したエージェントを削除する

[形式]

DelAgtSet2 (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	Agt	エージェント変数

DuplicateAgtSet

エージェント集合型変数2を、エージェント集合型変数1にコピーする

[形式]

DuplicateAgtSet (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2

Forward

カッコ内の数だけそのエージェントが向いている方向へ進む(進行方向はDirection変数による)。戻り値は、正常終了のときは-1、空間がループしていないときで、指定した距離を進めなかった時は、進めなかった距離を返す

[形式]

Forward(arg1)

[引数]

引数名	型	説明
arg1	Double	距離

[戻り値]

型	説明
Double	終了値（正常終了時は-1、進めなかったときは進めなかった距離）

ForwardDirectionCell

セル上を指定した方向へ移動する

[形式]

ForwardDirectionCell(arg1, arg2)

[引数]

引数名	型	説明
arg1	Integer	//方向 <原点が左下の場合> ※格子モデルのとき、0:右、1:右上、2:上、3:左上、4:左、5:左下、6:下、7:右下 ※六角モデルのとき、0:右、1:右上、2:左上、3:左、4:左下、5:右下 <原点が左上の場合> ※格子モデルのとき、0:右、1:右下、2:下、3:左下、4:左、5:左上、6:上、7:右上 ※六角モデルのとき、0:右、1:右下、2:左下、3:左、4:左上、5:右上
arg2	Integer	距離

[戻り値]

型	説明
Integer	終了値（正常終了時は-1、引数が不正なときは-2、指定した距離を進めなかったときは進めなかった距離）

ForwardX

カッコ内の数だけX軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す

[形式]

ForwardX(arg1)

[引数]

引数名	型	説明
arg1	Double	距離

[戻り値]

型	説明
Double	終了値（正常終了時は-1、進めなかったときは進めなかった距離）

ForwardXCell

セル上をカッコ内の数だけX軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す

[形式]

ForwardXCell (arg1)

[引数]

引数名	型	説明
arg1	Integer	距離

[戻り値]

型	説明
Integer	終了値（正常終了時は-1、進めなかったときは進めなかった距離）

ForwardY

カッコ内の数だけY軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す

[形式]

ForwardY (arg1)

[引数]

引数名	型	説明
arg1	Double	距離

[戻り値]

型	説明
Double	終了値（正常終了時は-1、進めなかったときは進めなかった距離）

ForwardYCell

セル上をカッコ内の数だけY軸方向に進む。戻り値は、正常終了のときは-1、空間がループしていないときで指定した距離を進めなかったときは進めなかった距離を返す

[形式]

ForwardYCell (arg1)

[引数]

引数名	型	説明
arg1	Integer	距離

[戻り値]

型	説明
Integer	終了値（正常終了時は-1、進めなかったときは進めなかった距離）

GetAgt

エージェント集合型変数の指定位置(0以上の整数)にあるエージェントを取得する

[形式]

GetAgt (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Integer	取得位置

[戻り値]

型	説明
Agt	エージェント型変数 (取得に失敗したときは-1)

GetAgtEntry

エージェント集合型変数に指定エージェントが含まれているかを調査する。指定エージェントが含まれている場合、エージェント集合型変数の先頭からの連番(要素番号)のうち最も小さい値を0以上の整数で返す。含まれていなかった場合は-1を返す

[形式]

GetAgtEntry (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Agt	調査したいエージェント型変数

[戻り値]

型	説明
Integer	要素番号 (エージェントが存在しない場合は-1)

GetDirection

A地点からB地点への角度を求める。空間がループする場合は、最短距離として求められる方向の角度を求める

[形式]

GetDirection (arg1, arg2, arg3, arg4, arg5)

[引数]

引数名	型	説明
arg1	Double	A地点のX座標
arg2	Double	A地点のY座標
arg3	Double	B地点のX座標
arg4	Double	B地点のY座標
arg5	Space	空間

[戻り値]

型	説明
Double	角度

GetHeightSpaceOwn

エージェント自身が乗っている空間の縦幅を取得する

[形式]

GetHeightSpaceOwn()

[戻り値]

型	説明
Integer	空間の縦幅

GetHistory

記憶した変数値を取得する

[形式]

GetHistory(arg1, arg2)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, Long, String, Agt	変数名
arg2	Integer	履歴番号(過去のある状態へ、現在から遡るステップ数)

[戻り値]

型	説明
Boolean, Double, Integer, Long, String, Agt	変数の値 (存在しない過去の値を参照したとき、ブール型変数はFalse、文字列型変数は""、整数型変数は0、長整数型変数は0、実数型変数は0.0を返す)

GetUniqueID

エージェントIDからUniqueIDを取得する

[形式]

GetUniqueID(arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtType	エージェント種別型変数
arg2	Integer	エージェントID

[戻り値]

型	説明
Agt	エージェントのUniqueID

GetWidthSpaceOwn

エージェント自身が乗っている空間の横幅を取得する

[形式]

GetWidthSpaceOwn()

[戻り値]

型	説明
Integer	空間の横幅

JoinAgtSet

エージェント集合型変数1にエージェント集合型変数2を追加する(要素の重複を認める)

[形式]

JoinAgtSet(arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2

KillAgt

エージェントを削除する(エージェントが実際に削除されるのは、ステップの終了時)

[形式]

KillAgt(arg1)

[引数]

引数名	型	説明
arg1	Agt	エージェント型変数

MakeAgtSet

指定したエージェント種別のエージェント全てを、エージェント集合型変数に格納する

[形式]

MakeAgtSet(arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	生成されるエージェント集合型変数
arg2	AgtType	エージェント種別型変数

MakeAgtSetAroundOwnCell

指定したエージェント集合の中から、エージェント自身の周り(セル上)にいるエージェントを選び、エージェント集合型変数に格納する

[形式]

MakeAgtSetAroundOwnCell (arg1, arg2, arg3, arg4)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Integer	視野
arg3	AgtSet	指定したエージェント集合型変数
arg4	Boolean	エージェント自身の存在フラグ (エージェント自身を含めるときはTrue、含めないときはFalse)

MakeAgtSetAroundPositionCell

指定したエージェント集合の中から、指定した位置座標の周り(セル上)にいるエージェントを選び、指定したエージェント集合型変数内に格納する

[形式]

MakeAgtSetAroundPositionCell (arg1, arg2, arg3, arg4, arg5, arg6, arg7)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Space	空間
arg3	Integer	X座標
arg4	Integer	Y座標
arg5	Integer	Layer
arg6	Integer	視野
arg7	AgtSet	指定したエージェント集合型変数

MakeAgtSetSpace

指定した空間上に存在する全てのエージェントを指定したエージェント集合型変数に格納する

[形式]

MakeAgtSetSpace (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Space	空間

MakeAllAgtSetAroundOwn

エージェント自身の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野の範囲の周囲にいる全てのエージェントをリストアップ)

[形式]

MakeAllAgtSetAroundOwn (arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Double	視野
arg3	Boolean	エージェント自身の存在フラグ (エージェント自身を含めるときはTrue、含めないときはFalse)

MakeAllAgtSetAroundOwnCell

エージェント自身の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野(格子)の範囲の周囲にいる全てのエージェントをリストアップ)

[形式]

MakeAllAgtSetAroundOwnCell(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Integer	視野
arg3	Boolean	エージェント自身の存在フラグ (エージェント自身を含めるときはTrue、含めないときはFalse)

MakeAllAgtSetAroundPosition

指定した位置座標の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、指定した位置座標の周囲にいる全てのエージェントをリストアップ)

[形式]

MakeAllAgtSetAroundPosition(arg1, arg2, arg3, arg4, arg5, arg6)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Space	空間
arg3	Double	X座標
arg4	Double	Y座標
arg5	Integer	Layer
arg6	Double	視野

MakeAllAgtSetAroundPositionCell

セル上に指定した位置座標の周り(セル上)にいる全てのエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、セル上に指定した位置座標の周囲にいる全てのエージェントをリストアップ)

[形式]

MakeAllAgtSetAroundPositionCell(arg1, arg2, arg3, arg4, arg5, arg6)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数

arg2	Space	空間
arg3	Integer	X座標
arg4	Integer	Y座標
arg5	Integer	Layer
arg6	Integer	視野

MakeCommonAgtSet

エージェント集合型変数2とエージェント集合型変数3のどちらにも含まれているエージェントからなるエージェント集合型変数1を取得する

[形式]

MakeCommonAgtSet(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2
arg3	AgtSet	エージェント集合型変数3

MakeDiffAgtSet

エージェント集合型変数2とエージェント集合型変数3のどちらか一方にだけ含まれているエージェントからなるエージェント集合型変数1を取得する

[形式]

MakeDiffAgtSet(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2
arg3	AgtSet	エージェント集合型変数3

MakeOneAgtSetAroundOwn

エージェント自身の周りにおける指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野の範囲の周囲にいる指定種別のエージェントをリストアップ)

[形式]

MakeOneAgtSetAroundOwn(arg1, arg2, arg3, arg4)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Double	視野
arg3	AgtType	エージェント種別型変数
arg4	Boolean	エージェント自身の存在フラグ(エージェント自身を含めるときはTrue、含まないときはFalse)

MakeOneAgtSetAroundOwnCell

エージェント自身の周り(セル上)にいる指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、視野(格子)の範囲の周囲にいる指定種別のエージェントをリストアップ)

[形式]

MakeOneAgtSetAroundOwn(arg1, arg2, arg3, arg4)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Integer	視野
arg3	AgtType	エージェント種別型変数
arg4	Boolean	エージェント自身の存在フラグ (エージェント自身を含めるときはTrue、含めないときはFalse)

MakeOneAgtSetAroundPosition

指定した位置座標の周り(セル上)にいる指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、指定した位置座標の周囲にいる指定種別のエージェントをリストアップ)

[形式]

MakeOneAgtSetAroundPosition(arg1, arg2, arg3, arg4, arg5, arg6, arg7)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Space	空間
arg3	Double	X座標
arg4	Double	Y座標
arg5	Integer	Layer
arg6	Double	視野
arg7	AgtType	エージェント種別型変数

MakeOneAgtSetAroundPositionCell

セル上に指定した位置座標の周り(セル上)にいる指定種別のエージェントのエージェント集合型変数を取得する(エージェント集合型変数に、セル上に指定した位置座標の周囲にいる指定種別のエージェントをリストアップ)

[形式]

MakeOneAgtSetAroundPositionCell(arg1, arg2, arg3, arg4, arg5, arg6, arg7)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Space	空間
arg3	Integer	X座標
arg4	Integer	Y座標

arg5	Integer	Layer
arg6	Double	視野
arg7	AgtType	エージェント種別型変数

MergeAgtSet

エージェント集合型変数1にエージェント集合型変数2を追加する(要素の重複は認めない)

[形式]

MergeAgtSet(arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2

MoveLayerSpace

移動させたいLayer数だけLayerを移動させる

[形式]

MoveLayerSpace(arg1)

[引数]

引数名	型	説明
arg1	Integer	移動させたいLayer数

[戻り値]

型	説明
Integer	終了値(正常終了時は-1、Layerの最上段もしくは再下段に到達して進めなかったときはそのLayer数)

MoveToCenter

エージェント自身が乗っている空間(Layer)の中央に移動する

[形式]

MoveToCenter()

MoveToSpaceAgtSetCell

指定された座標を中心に、指定エージェント集合型変数に属するエージェントには重ならないように周りの空き地(セル)を探し、移動する。C

[形式]

MoveToSpaceAgtSetCell(arg1, arg2, arg3, arg4, arg5, arg6)

[引数]

引数名	型	説明
-----	---	----

arg1	Space	空間
arg2	Integer	X座標
arg3	Integer	Y座標
arg4	Integer	Layer
arg5	Integer	視野
arg6	AgtSet	エージェント集合型変数

[戻り値]

型	説明
Boolean	移動エラー値 (移動できなかったときはTrue、できたときはFalse)

MoveToSpaceOwnCell

エージェント自身の周りの空き地 (セル) を探し、移動する。(検索された領域に複数の空き地が存在するときは、ランダムに移動位置を決定)

[形式]

MoveToSpaceOwnCell (arg1)

[引数]

引数名	型	説明
arg1	Integer	視野

[戻り値]

型	説明
Boolean	移動エラー値 (移動できなかったときはTrue、できたときはFalse)

MoveToSpacePositionCell

指定された座標を中心に、周りの空き地 (セル) を探し、移動する。(検索された領域に複数の空き地が存在するときは、ランダムに移動位置を決定)

[形式]

MoveToSpaceAgtSetCell (arg1, arg2, arg3, arg4, arg5)

[引数]

引数名	型	説明
arg1	Space	空間
arg2	Integer	X座標
arg3	Integer	Y座標
arg4	Integer	Layer
arg5	Integer	視野

[戻り値]

型	説明
Boolean	移動エラー値 (移動できなかったときはTrue、できたときはFalse)

OnLayerSpace

指定したLayerへ移動する

[形式]

OnLayerSpace (arg1)

[引数]

引数名	型	説明
arg1	Integer	移動先のLayer数

[戻り値]

型	説明
Integer	終了値 (正常終了時は-1、移動できなかったときは0)

PickupAgt

エージェント集合型変数の指定位置 (0以上の整数)にあるエージェントを取得し、取得したエージェントをエージェント集合型変数から削除する

[形式]

PickupAgt (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Integer	取得位置

[戻り値]

型	説明
Agt	エージェント型変数 (取得に失敗したときは-1)

PurifyAgtSet

エージェント集合型変数2の中からエージェントの重複を取り除いたエージェント集合型変数1を取得する

[形式]

PurifyAgtSet (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数1
arg2	AgtSet	エージェント集合型変数2

Pursue

目標としたエージェントに向かって移動する。移動方向に従い、Direction変数の値が更新される

[形式]

Pursue (arg1, arg2)

[引数]

引数名	型	説明
arg1	Agt	目標となるエージェント
arg2	Double	移動距離

[戻り値]

型	説明
Double	終了値（正常終了時は-1、進めなかったときは進めなかった距離）

RandomPutAgtSet

指定されたエージェント集合をランダムに配置する

[形式]

RandomPutAgtSet (arg1)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数

RandomPutAgtSetCell

指定されたエージェント集合をセル上にランダムに配置する

[形式]

RandomPutAgtSetCell (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Boolean	他のエージェントとの重なり（重なって良いときはTrue、重ならないときはFalse）

[戻り値]

型	説明
Integer	終了値（正常終了のとき-1、他のエージェントとの重なりがFalseであるにも関わらずエージェントの数が空間サイズより多いことにより重なってしまったときは、重なったエージェント数）

RemoveAgt

エージェント集合型変数から指定エージェントを削除する

[形式]

RemoveAgt (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	Agt	エージェント型変数

ReverseDirectionCell

セル上で逆方向を取得する

[形式]

ReverseDirectionCell (arg1)

[引数]

引数名	型	説明
arg1	Integer	//方向 <原点が左下の場合> ※格子モデルのとき、0:右、1:右上、2:上、3:左上、4:左、5:左下、6:下、7:右下 ※六角モデルのとき、0:右、1:右上、2:左上、3:左、4:左下、5:右下 <原点が左上の場合> ※格子モデルのとき、0:右、1:右下、2:下、3:左下、4:左、5:左上、6:上、7:右上 ※六角モデルのとき、0:右、1:右下、2:左下、3:左、4:左上、5:右上

[戻り値]

型	説明
Integer	//逆方向 <原点が左下の場合> ※格子モデルのとき、0:右、1:右上、2:上、3:左上、4:左、5:左下、6:下、7:右下 ※六角モデルのとき、0:右、1:右上、2:左上、3:左、4:左下、5:右下 <原点が左上の場合> ※格子モデルのとき、0:右、1:右下、2:下、3:左下、4:左、5:左上、6:上、7:右上 ※六角モデルのとき、0:右、1:右下、2:左下、3:左、4:左上、5:右上 ※引数が不正なときは-2

SortAgtSet

エージェント集合型変数を指定された変数をキーとしてソートする

[形式]

SortAgtSet (arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	AgtSet	エージェント集合型変数
arg2	String	ソートのキーとなる変数を表す文字列
arg3	Boolean	ソート順 (昇順のときはTrue、降順のときはFalse)

SpecifyAgtType

指定したエージェントのエージェント種別を取得する

[形式]

SpecifyAgtType (arg1)

[引数]

引数名	型	説明
arg1	Agt	エージェント型変数

[戻り値]

型	説明
AgtType	エージェント種別型変数

SpecifyKillAgt

Killフラグの値を取得する

[形式]

SpecifyKillAgt (arg1)

[引数]

引数名	型	説明
arg1	Agt	エージェント型変数

[戻り値]

型	説明
Boolean	判定値 (KillフラグがOnのときはTrue、OffのときはFalse)

TerminateAgt

エージェントを完全削除する ※エージェントはルール実行中であっても即座に削除される ※エージェント型変数のうち、該当エージェントが含まれる変数は-1が代入される ※エージェント集合型変数に該当エージェントが含まれる場合は、エージェント集合型変数からも削除される

[形式]

TerminateAgt (arg1)

[引数]

引数名	型	説明
arg1	Agt	エージェント型変数

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

Turn

向きを変える (Direction変数に反映)。引数が正の値のときは左回り、負の値のときは右回りに向きを変える (左下原点の場合)。※この関数によりDirection変数は0?360の値を取る ※左上原点の設定時は回転方向が逆になる。

[形式]

Turn (arg1)

[引数]

引数名	型	説明
arg1	Double	角度 (角度はdegree(°)で指定する)

TurnAgt

指定されたエージェントの方向を向く

[形式]

TurnAgt (arg1)

[引数]

引数名	型	説明
arg1	Double	指定エージェント

名称	説明
GetHeightSpace	指定された空間の縦幅を取得する
GetHeightSpaceOwn	エージェント自身が乗っている空間の縦幅を取得する
GetLayerSpace	指定された空間のLayer数を取得する
GetRideSpace	指定したエージェント種別が乗っている空間を取得する
GetWidthSpace	指定された空間の横幅を取得する
MeasureDistance	2点間の最短距離を測定する
SpecifyLoop	指定した空間のループ設定を取得する
SpecifySpace	指定した空間の種別を取得する

GetHeightSpace

指定された空間の縦幅を取得する

[形式]

GetHeightSpace (arg1)

[引数]

引数名	型	説明
arg1	Space	空間

[戻り値]

型	説明
Integer	空間の縦幅

GetHeightSpaceOwn

エージェント自身が乗っている空間の縦幅を取得する

[形式]

GetHeightSpaceOwn ()

[戻り値]

型	説明
Integer	空間の縦幅

GetLayerSpace

指定された空間のLayer数を取得する

[形式]

GetLayerSpace (arg1)

[引数]

引数名	型	説明
arg1	Space	空間

[戻り値]

型	説明
Integer	空間のLayer数

GetRideSpace

指定したエージェント種別が乗っている空間を取得する

[形式]

GetRideSpace (arg1)

[引数]

引数名	型	説明
arg1	AgtType	エージェント種別型変数

[戻り値]

型	説明
Space	空間

GetWidthSpace

指定された空間の横幅を取得する

[形式]

GetWidthSpace (arg1)

[引数]

引数名	型	説明
arg1	Space	空間

[戻り値]

型	説明
Integer	空間の横幅

MeasureDistance

2点間の最短距離を測定する

[形式]

MeasureDistance (arg1, arg2, arg3, arg4, arg5)

[引数]

引数名	型	説明
arg1	Double	X座標1
arg2	Double	Y座標1
arg3	Double	X座標2

arg4	Double	Y座標2
arg5	Space	空間

[戻り値]

型	説明
Double	2点間の最短距離

SpecifyLoop

指定した空間のループ設定を取得する

[形式]

SpecifyLoop(arg1)

[引数]

引数名	型	説明
arg1	Space	空間

[戻り値]

型	説明
Boolean	判定値 (ループするときはTrue、しないときはFalse)

SpecifySpace

指定した空間の種別を取得する

[形式]

SpecifySpace(arg1)

[引数]

引数名	型	説明
arg1	Space	空間

[戻り値]

型	説明
Integer	空間種別 (格子モデルは0、六角モデルは1)

名称	説明
CloseFileXML	XMLファイルを閉じる ※SetXMLValue関数で変更された値がXMLファイルに反映される
DeleteXMLDom	XMLデータを廃棄する
FlushXMLDom	XMLデータへの操作を確定する
GetXMLValue	XMLデータから値を取得する
MakeXMLDom	指定した文字列変数をXMLデータとして定義する
OpenFileXML	XMLファイルを開く
RemoveXMLValue	XMLデータ識別番号で指定されたXMLデータのうち、XPath文字列で指定したノードを削除する。CloseFileXML関数もしくはFlushXMLDom関数を発行した時点で値が確定される
SetXMLValue	XMLデータから値を代入する。CloseFileXML関数もしくはFlushXMLDom関数を発行した時点で値が確定される

CloseFileXML

XMLファイルを閉じる ※SetXMLValue関数で変更された値がXMLファイルに反映される

[形式]

CloseFileXML (arg1)

[引数]

引数名	型	説明
arg1	Integer	閉じたいXMLデータ識別番号

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

DeleteXMLDom

XMLデータを廃棄する

[形式]

DeleteXMLDom (arg1)

[引数]

引数名	型	説明
arg1	Integer	廃棄したいXMLデータ識別番号

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

FlushXMLDom

XMLデータへの操作を確定する

[形式]

FlushXMLDom(arg1)

[引数]

引数名	型	説明
arg1	Integer	操作を確定したいXMLデータ識別番号

GetXMLValue

XMLデータから値を取得する

[形式]

GetXMLValue(arg1, arg2)

[引数]

引数名	型	説明
arg1	Integer	値を取得したいXMLデータ識別番号
arg2	String	XPath文字列

[戻り値]

型	説明
String	文字列

MakeXMLDom

指定した文字列変数をXMLデータとして定義する

[形式]

MakeXMLDom(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	文字列型変数
arg2	Integer	XMLデータ識別番号の指定 (1以上の整数)

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

OpenFileXML

XMLファイルを開く

[形式]

OpenFileXML(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	開きたいファイルの名称
arg2	Integer	XMLデータ識別番号の指定 (1以上の整数)

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

RemoveXMLValue

XMLデータ識別番号で指定されたXMLデータのうち、XPath文字列で指定したノードを削除する。CloseFileXML関数もしくはFlushXMLDom関数を発行した時点で値が確定される

[形式]

RemoveXMLValue(arg1, arg2)

[引数]

引数名	型	説明
arg1	Integer	XMLデータ識別番号
arg2	String	XPath文字列

SetXMLValue

XMLデータから値を代入する。CloseFileXML関数もしくはFlushXMLDom関数を発行した時点で値が確定される

[形式]

SetXMLValue(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Integer	値を代入したいXMLデータ識別番号
arg2	String	XPath文字列
arg3	String	代入する値

名称	説明
CloseFile	テキストファイルを閉じる
CloseFileCSV	CSVファイルを閉じる
IsEofFile	テキストファイル読み込みの最終行を判定する
OpenFile	テキストファイルを開く
OpenFileCSV	CSVファイルを開く
OpenFileCode	テキストファイルを文字コードを指定して開く
ReadFile	テキストファイルから行を読み込む
ReadFileCSV	CSVファイルから1データを読み込む ※読み込み形式は、ダブルクォーテーション(“”)に対応していない
WriteFile	テキストファイルへ1データを書き込む
WriteFileCSV	CSVファイルへ1データを書き込む
WriteLnFile	テキストファイルへ改行つきで1データを書き込む

CloseFile

テキストファイルを閉じる

[形式]

CloseFile (arg1)

[引数]

引数名	型	説明
arg1	Integer	閉じたいファイル識別番号の指定

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

CloseFileCSV

CSVファイルを閉じる

[形式]

CloseFileCSV (arg1)

[引数]

引数名	型	説明
arg1	Integer	閉じたいファイル識別番号の指定

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

IsEofFile

テキストファイル読み込みの最終行を判定する

[形式]

IsEofFile (arg1)

[引数]

引数名	型	説明
arg1	Integer	判定したいファイル識別番号の指定

[戻り値]

型	説明
Boolean	判定値 (最終行のときはTrue、最終行でないときはFalse)

OpenFile

テキストファイルを開く

[形式]

OpenFile (arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	String	開きたいファイルの名称
arg2	Integer	ファイル識別番号の指定 (1以上の整数)
arg3	Integer	オープンモード (Readのときは1:ファイル内データの読み込み目的で開く、Writeのときは2:ファイル内データを破棄してゼロからデータを書き出す目的で開く、Appendのときは3:ファイル内データを保持して最終行に追記する目的で開く)

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

OpenFileCSV

CSVファイルを開く

[形式]

OpenFileCSV (arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	String	開きたいファイルの名称
arg2	Integer	ファイル識別番号の指定 (1以上の整数)
arg3	Integer	オープンモード (Readのときは1、Writeのときは2、Appendのときは3)

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

OpenFileCode

テキストファイルを文字コードを指定して開く

[形式]

OpenFileCode(arg1, arg2, arg3, arg4)

[引数]

引数名	型	説明
arg1	String	開きたいファイルの名称
arg2	Integer	ファイル識別番号の指定 (1以上の整数)
arg3	Integer	オープンモード (Readのときは1、Writeのときは2、Appendのときは3)
arg4	String	blc]、"ISO-2022-JP")

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

ReadFile

テキストファイルから行を読み込む

[形式]

ReadFile(arg1)

[引数]

引数名	型	説明
arg1	Integer	読み込みたいファイル識別番号

[戻り値]

型	説明
String	読み込んだ文字列

ReadFileCSV

CSVファイルから1データを読み込む ※読み込み形式は、ダブルクォーテーション(" ")に対応していない

[形式]

ReadFileCSV(arg1)

[引数]

引数名	型	説明
arg1	Integer	読み込みたいファイル識別番号

[戻り値]

型	説明
String	読み込んだ文字列

WriteFile

テキストファイルへ1データを書き込む

[形式]

WriteFile(arg1, arg2)

[引数]

引数名	型	説明
arg1	Integer	書き出したいファイル識別番号
arg2	Boolean, Double, Integer, Long, String	書き出す内容

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

WriteFileCSV

CSVファイルへ1データを書き込む

[形式]

WriteFileCSV(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Integer	書き出したいファイル識別番号
arg2	Boolean, Double, Integer, Long, String	書き出す内容
arg3	Boolean	改行フラグ (Trueのときは出力後に改行あり、Falseのときは改行せずに「,」を追加)

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

WriteLnFile

テキストファイルへ改行つきで1データを書き込む

[形式]

WriteLnFile(arg1, arg2)

[引数]

引数名	型	説明
arg1	Integer	書き出したいファイル識別番号
arg2	Boolean, Double, Integer, Long, String	書き出す内容

[戻り値]

型	説明
Boolean	判定値 (成功のときはTrue、失敗のときはFalse)

名称	説明
InputBox	入力ダイアログを表示する。※入力が終了するまで、シミュレーションは停止する
MsgBox	ダイアログを表示する。※入力が終了するまで、シミュレーションは停止する
MsgDialog	ダイアログを表示する。※入力にかかわらず、シミュレーションは実行し続ける

InputBox

入力ダイアログを表示する。※入力が終了するまで、シミュレーションは停止する

[形式]

InputBox (arg1, arg2)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, Long, String	入力する変数名
arg2	String	表示メッセージ (表示メッセージの文字列中に「¥n」を指定すると文章を改行する)

MsgBox

ダイアログを表示する。※入力が終了するまで、シミュレーションは停止する

[形式]

MsgBox (arg1)

[引数]

引数名	型	説明
arg1	String	表示メッセージ (表示メッセージの文字列中に「¥n」を指定すると文章を改行する)

MsgDialog

ダイアログを表示する。※入力にかかわらず、シミュレーションは実行し続ける

[形式]

MsgDialog (arg1)

[引数]

引数名	型	説明
arg1	String	表示メッセージ (表示メッセージの文字列中に「¥n」を指定すると文章を改行する)

名称	説明
CloseDB	開いているデータベースを閉じる (※Professional Only)
CommitDB	データベースをコミットする (※Professional Only) ※データベースにおける一連の更新処理が全て正常に完結したことを宣言 (処理を確定して、データベースを更新) する
GetValueDB	指定したフィールドの値を取得する (※Professional Only) ※QueryDB関数の実行後に実行可能
MakeXMLDomSelectDB	SQLクエリーを発行して取得した値をXMLデータとして保持する (※Professional Only)
NextDB	直前に実行したQueryDB関数の指定内容に合致するレコードに関して以下の内容を実行する (※Professional Only) ※QueryDB関数が成功した場合のみ実行可能 ※初回の実行では先頭のレコードを参照する。※実行毎に参照レコード番号を1つ進める。※参照先のレコードのデータが存在する場合はTrue,存在しない場合はFalseを返す。
OpenDB	データベースを開く (※Professional Only) ※対応データベースは、PostgreSQL 8.1/8.2、SQL Server 2000/2005
QueryDB	開いているデータベースに対して、指定したSQLクエリーを発行する (※Professional Only) ※データベースの変更が伴うときは、CommitDB関数もしくはRollbackDB関数を実行する
RollbackDB	データベースをロールバックする (※Professional Only)
UpdateDB	開いているデータベースに対して、指定したSQLクエリーを発行する。(※Professional Only) ※ UpdateDB関数はUPDATE文、CREATE文、DROP文、INSERT文、DELETE文を用いる場合に使用し、SELECT文を使用する場合はQueryDB関数を使用する ※ データベースの変更が伴うときは、CommitDB関数もしくはRollbackDB関数を実行する。

CloseDB

開いているデータベースを閉じる (※Professional Only)

[形式]

CloseDB()

[戻り値]

型	説明
Boolean	実行結果 (成功したときはTrue、失敗したときはFalse)

CommitDB

データベースをコミットする (※Professional Only) ※データベースにおける一連の更新処理が全て正常に完結したことを宣言 (処理を確定して、データベースを更新) する

[形式]

CommitDB()

GetValueDB

指定したフィールドの値を取得する (※Professional Only) ※QueryDB関数の実行後に実行可能

[形式]

GetValueDB (arg1)

[引数]

引数名	型	説明
arg1	String	フィールド名

[戻り値]

型	説明
String	指定したフィールドの値

MakeXMLDomSelectDB

SQLクエリーを発行して取得した値をXMLデータとして保持する (※Professional Only)

[形式]

MakeXMLDomSelectDB (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	SQL文字列
arg2	Integer	XMLデータ識別番号の指定 (1以上の整数)

[戻り値]

型	説明
Integer	レコード件数

NextDB

直前に実行したQueryDB関数の指定内容に合致するレコードに関して以下の内容を実行する (※Professional Only) ※QueryDB関数が成功した場合のみ実行可能 ※初回の実行では先頭のレコードを参照する。 ※実行毎に参照レコード番号を1つ進める。 ※参照先のレコードのデータが存在する場合はTrue,存在しない場合はFalseを返す。

[形式]

NextDB ()

[戻り値]

型	説明
Boolean	実行結果 (次のレコードが存在するときはTrue、存在しないときはFalse)

OpenDB

データベースを開く (※Professional Only) ※対応データベースは、PostgreSQL 8.1/8.2、SQL Server 2000/2005

[形式]

OpenDB (arg1, arg2, arg3, arg4, arg5)

[引数]

引数名	型	説明
arg1	String	JDBC URL (PostgreSQL 8.1/8.2のとき「jdbc:postgresql://localhost:5432」、SQL Server 2000/2005のとき「jdbc:sqlserver://localhost:1433;DatabaseName=データベース名」)
arg2	String	JDBC ドライバ名 (PostgreSQL 8.1/8.2のとき「org.postgresql.Driver」、SQL Server 2000/2005のとき「com.microsoft.sqlserver.jdbc.SQLServerDriver」)
arg3	String	データベース名 (PostgreSQL 8.1/8.2のとき「データベース名」、SQL Server 2000/2005のとき「”」)
arg4	String	データベースユーザ名
arg5	String	パスワード

[戻り値]

型	説明
Boolean	接続結果 (成功のときはTrue、失敗のときはFalse)

QueryDB

開いているデータベースに対して、指定したSQLクエリーを発行する (※Professional Only) ※データベースの変更が伴うときは、CommitDB関数もしくはRollbackDB関数を実行する

[形式]

QueryDB(arg1)

[引数]

引数名	型	説明
arg1	String	SQL文字列

[戻り値]

型	説明
Integer	実行結果 (成功のときは0以上の整数、失敗のときは-1以下の整数)

RollbackDB

データベースをロールバックする (※Professional Only)

[形式]

RollbackDB()

UpdateDB

開いているデータベースに対して、指定したSQLクエリーを発行する。(※Professional Only) ※ UpdateDB関数はUPDATE文、CREATE文、DROP文、INSERT文、DELETE文を用いる場合に使用し、SELECT文を使用する場合はQueryDB関数を使用する ※ データベースの変更が伴うときは、CommitDB関数もしくはRollbackDB関数を実行する。

[形式]

UpdateDB()

[引数]

引数名	型	説明
-----	---	----

arg1	String	SQL文字列
------	--------	--------

[戻り値]

型	説明
Integer	実行結果 (成功のときは0以上の整数、失敗のときは-1以下の整数)

名称	説明
DeliverRemoteInfo	グローバル空間に関する設定情報を、リモート設定内のリモート識別名リストに登録された各宛先に配信する (※Professional Only) ※Univ_Initでのみ実行可能
GetLocalX	グローバル空間のX座標をローカル空間のX座標に変換する (※Professional Only) ※Univ_InitとAgt_Init以外で実行可能
GetLocalY	グローバル空間のY座標をローカル空間のY座標に変換する (※Professional Only) ※Univ_InitとAgt_Init以外で実行可能
GetRemoteArray	リモート配列変数の値を取得する ※Univ_InitとAgt_Init以外で実行可能
GetRemoteName	リモート識別名を取得する (※Professional Only) ※Univ_InitとAgt_Init以外で実行可能
GetRemoteValue	リモート変数の値を取得します (※Professional Only) ※Univ_InitとAgt_Init以外で実行可能
GoogleSearch	Googleで検索を行う (※Professional Only)
MakeHtmlDom	指定したURLからHTMLテキストを取り込み、XMLデータとして格納する (※Professional Only)
RegistHtmlPage	指定した相対URLパスへのリクエストに対して、表示するHTMLを格納する変数を指定する ※事前にWebStart関数を実行しておく (※Professional Only)
RegistHtmlPageToXML	指定した相対URLパスへのリクエストに対して、表示するHTMLを格納する変数を指定し、戻り値をXMLデータとして保存する。戻り値のXMLデータは次の通り <HttpSet> <Http> <パラメータ名>値</パラメータ名> </Http> </HttpSet> ※事前にWebStart関数を実行しておく (※Professional Only)
RemoteFinish	リモート設定で定義されるリモートマシンを全て停止する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)
RemoteInitialize	リモート設定で定義されるリモートマシンに対して初期化要求を送信する。リモートマシンが実行中の場合は一度停止してから再起動し、リモートマシンが停止中の場合は実行する。 ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)
RemoteName	自身のリモート識別名を定義する ※Univ_Initでのみ実行可能 (※Professional Only)
RemoteStep	リモート識別名で定義した相手先の実行を定義する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)
RemoteSyncStep	リモート設定で定義した相手先の同期モードでのステップ実行を行う。定義した相手先の実行が全て完了、もしくは定義したタイムアウトするまでの時間が経過したときに処理を完了する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)
SetRemoteArray	リモート配列変数の値を代入する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)
SetRemoteValue	リモート変数の値を代入する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)
TeleportationAgt	エージェントを他のartisocモデル上へ移動する (空間上に定義されているエージェントの移動は不可) ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)
WebLock	RegistHtmlPageToXML関数で定義した戻り値のXMLデータを保持する文字列変数の値を更新不可能にする (※Professional Only)
WebStart	Webサーバを起動する。シミュレーションの実行が終了するとWebサーバは停止する ※Univ_Initでのみ実行可能 (※Professional Only)
WebUnLock	RegistHtmlPageToXML関数で定義した戻り値のXMLデータを保持する文字列変数の値を更新可能にする (※Professional Only)

DeliverRemoteInfo

グローバル空間に関する設定情報を、リモート設定内のリモート識別名リストに登録された各宛先に配信する（※Professional Only）※Univ_Initでのみ実行可能

[形式]

DeliverRemoteInfo()

[戻り値]

型	説明
Integer	判定値（正常のときは0、エラーのときは0以外）

GetLocalX

グローバル空間のX座標をローカル空間のX座標に変換する（※Professional Only）※Univ_InitとAgt_Init以外で実行可能

[形式]

GetLocalX(arg1)

[引数]

引数名	型	説明
arg1	Double	グローバル空間のX座標

[戻り値]

型	説明
Double	ローカル空間のX座標

GetLocalY

グローバル空間のY座標をローカル空間のY座標に変換する（※Professional Only）※Univ_InitとAgt_Init以外で実行可能

[形式]

GetLocalY(arg1)

[引数]

引数名	型	説明
arg1	Double	グローバル空間のX座標

[戻り値]

型	説明
Double	ローカル空間のX座標

GetRemoteArray

リモート配列変数の値を取得する ※Univ_InitとAgt_Init以外で実行可能

[形式]

GetRemoteArray(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, Long, String	配列変数
arg2	String	接続する相手先のリモート識別名
arg3	String	リモート配列変数のパス名

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

GetRemoteName

リモート識別名を取得する (※Professional Only) ※Univ_InitとAgt_Init以外で実行可能

[形式]

GetRemoteName ()

[戻り値]

型	説明
String	リモート識別名

GetRemoteValue

リモート変数の値を取得します (※Professional Only) ※Univ_InitとAgt_Init以外で実行可能

[形式]

GetRemoteValue(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, Long, String	格納先の変数
arg2	String	接続する相手先のリモート識別名
arg3	String	リモート変数のパス名

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

GoogleSearch

Googleで検索を行う (※Professional Only)

[形式]

GoogleSearch(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	String	検索結果をXMLデータとして保持する文字列変数
arg2	String	Google Web APIのキーコード
arg3	String	検索文字列

[戻り値]

型	説明
Boolean	実行結果 (成功のときはTrue、失敗のときはFalse)

MakeHtmlDom

指定したURLからHTMLテキストを取り込み、XMLデータとして格納する (※Professional Only)

[形式]

MakeHtmlDom(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	String	URL文字列
arg2	String	HTMLデータを保持する文字列変数
arg3	Integer	XMLデータ識別番号の指定 (1以上の整数)

[戻り値]

型	説明
Boolean	実行結果 (成功のときはTrue、失敗のときはFalse)

RegistHtmlPage

指定した相対URLパスへのリクエストに対して、表示するHTMLを格納する変数を指定する ※事前にWebStart関数を実行しておく (※Professional Only)

[形式]

RegistHtmlPage(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	URL文字列
arg2	String	HTMLデータを保持する文字列変数名

[戻り値]

型	説明
String	実行結果 (正常のときはnull、エラーのときはエラーメッセージ)

RegistHtmlPageToXML

指定した相対URLパスへのリクエストに対して、表示するHTMLを格納する変数を指定し、戻り値をXMLデータとして

保存する。戻り値のXMLデータは次の通り <HttpSet> <Http> <パラメータ名>値</パラメータ名> </Http>
</HttpSet> ※事前にWebStart関数を実行しておく (※Professional Only)

[形式]

RegistHtmlPageToXML (arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	String	URL文字列
arg2	String	HTMLデータを保持する文字列変数名
arg3	String	戻り値のXMLデータを保持する文字列変数名

[戻り値]

型	説明
String	実行結果 (正常のときはnull、エラーのときはエラーメッセージ)

RemoteFinish

リモート設定で定義されるリモートマシンを全て停止する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)

[形式]

RemoteFinish ()

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

RemoteInitialize

リモート設定で定義されるリモートマシンに対して初期化要求を送信する。リモートマシンが実行中の場合は一度停止してから再起動し、リモートマシンが停止中の場合は実行する。 ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)

[形式]

RemoteInitialize ()

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

RemoteName

自身のリモート識別名を定義する ※Univ_Initでのみ実行可能 (※Professional Only)

[形式]

RemoteName (arg1, arg2)

[引数]

引数名	型	説明
-----	---	----

arg1	String	リモート識別名
arg2	String	同期モード (分析実行のための割り込み処理を待つときはTrue、待たないときはFalse)

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

RemoteStep

リモート識別名で定義した相手先の実行を定義する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)

[形式]

RemoteStep(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	相手先のリモート識別名
arg2	String	同期モード (リモートのステップ処理が完了するのを待つときはTrue、待たないときはFalse)

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

RemoteSyncStep

リモート設定で定義した相手先の同期モードでのステップ実行を行う。定義した相手先の実行が全て完了、もしくは定義したタイムアウトするまでの時間が経過したときに処理を完了する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)

[形式]

RemoteSyncStep(arg1)

[引数]

引数名	型	説明
arg1	Integer	タイムアウトするまでの時間 (ミリ秒)

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

SetRemoteArray

リモート配列変数の値を代入する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)

[形式]

SetRemoteArray(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, Long, String	配列変数
arg2	String	接続する相手先のリモート識別名
arg3	String	リモート配列変数のパス名

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

SetRemoteValue

リモート変数の値を代入する ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)

[形式]

SetRemoteValue(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Boolean, Double, Integer, Long, String	変数の値
arg2	String	接続する相手先のリモート識別名
arg3	String	リモート変数のパス名

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

TeleportationAgt

エージェントを他のartriosocモデル上へ移動する(空間上に定義されているエージェントの移動は不可) ※Univ_InitとAgt_Init以外で実行可能 (※Professional Only)

[形式]

TeleportationAgt(arg1, arg2)

[引数]

引数名	型	説明
arg1	Agt	エージェント型変数
arg2	String	移動する相手先のリモート識別名

[戻り値]

型	説明
Integer	判定値 (正常のときは0、エラーのときは0以外)

WebLock

RegistHtmlPageToXML関数で定義した戻り値のXMLデータを保持する文字列変数の値を更新不可能にする (※Professional Only)

[形式]

WebLock ()

WebStart

Webサーバを起動する。シミュレーションの実行が終了するとWebサーバは停止する ※Univ_Initでのみ実行可能 (※ Professional Only)

[形式]

WebStart (arg1)

[引数]

引数名	型	説明
arg1	Integer	ポート番号

WebUnlock

RegistHtmlPageToXML関数で定義した戻り値のXMLデータを保持する文字列変数の値を更新可能にする (※ Professional Only)

[形式]

WebUnlock ()

名称	説明
GetDayOfWeek	1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)から曜日を取得する
GetRealTime	1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)を取得する
Sleep	指定した時間(ミリ秒)だけシミュレーションの実行をスリープします
StrToTime	日時を表す文字列から1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)を取得する
TimeToStr	1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)から年月日時分秒の文字列を取得する

GetDayOfWeek

1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)から曜日を取得する

[形式]

GetDayOfWeek (arg1)

[引数]

引数名	型	説明
arg1	Long	1970年1月1日からのミリ秒

[戻り値]

型	説明
Integer	曜日 (0は日、1は月、2は火、3是水、4は木、5は金、6は土)

GetRealTime

1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)を取得する

[形式]

GetRealTime ()

[戻り値]

型	説明
Long	1970年1月1日からのミリ秒

Sleep

指定した時間(ミリ秒)だけシミュレーションの実行をスリープします

[形式]

Sleep (arg1)

[引数]

引数名	型	説明
arg1	Long	スリープするミリ秒

StrToTime

日時を表す文字列から1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)を取得する

[形式]

StrToTime (arg1)

[引数]

引数名	型	説明
arg1	String	日時文字列 (yyyy/MM/dd HH:mm:ss、もしくはyyyy/MM/ddで指定)

[戻り値]

型	説明
Long	1970年1月1日からの経過時間(ミリ秒)

TimeToStr

1970年1月1日グリニッジ標準時00:00:00.000からの経過時間(ミリ秒)から年月日時分秒の文字列を取得する

[形式]

TimeToStr (arg1, arg2)

[引数]

引数名	型	説明
arg1	Long	1970年1月1日からのミリ秒
arg2	String	出力形式 (yは年(西暦、yyyy、Mは月(MM指定)、dは日(dd指定)、Hは時間(HH指定)、mは分(mm指定)、sは秒(ss指定)と表示し、1桁のときは先頭を0で補完) ※「yyyy/MM/dd HH:mm:ss」、「y年M月d日」等と指定

[戻り値]

型	説明
String	年月日時分秒の文字列

名称	説明
Beep	ビーブ音を鳴らす
StartSound	サウンドファイル(*.wav、*.mid、*.mp3)を再生する
StopSound	サウンドファイルの再生を停止する

Beep

ビーブ音を鳴らす

[形式]

Beep ()

StartSound

サウンドファイル(*.wav、*.mid、*.mp3)を再生する

[形式]

StartSound (arg1)

[引数]

引数名	型	説明
arg1	String	サウンドファイルのパス名

StopSound

サウンドファイルの再生を停止する

[形式]

StopSound ()

名称	説明
AddOutputDefinition	出力設定画面を追加する
AgtTypeToPath	エージェント型変数の値からパス名を取得する
ChangeAgtRule	エージェントルールを変更する
ChangeOutputDefinition	出力設定を変更する
CreateAgtByString	パス名からエージェントを生成する
CreateAgtMulti	複数のエージェントを生成する
DefineAgtType	エージェント型を定義する
DefineSpace	空間を定義する
DefineVariable	新規に変数を定義する
SaveModel	モデルファイルを保存する (※Professional Only)
SpaceToPath	空間変数の値からパス名を取得する
TranscriptValue	エージェントが保持する変数の値を他のエージェントに転写する

AddOutputDefinition

出力設定画面を追加する

[形式]

AddOutputDefinition(arg1)

[引数]

引数名	型	説明
arg1	String	追加する出力画面の設定ファイル名

AgtTypeToPath

エージェント型変数の値からパス名を取得する

[形式]

AgtTypeToPath(arg1)

[引数]

引数名	型	説明
arg1	AgtType	エージェント型変数

[戻り値]

型	説明
String	パス名

ChangeAgtRule

エージェントルールを変更する

[形式]

ChangeAgtRule (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	エージェントのパス名
arg2	String	変更するエージェントルールのファイル名

ChangeOutputDefinition

出力設定を変更する

[形式]

ChangeOutputDefinition (arg1)

[引数]

引数名	型	説明
arg1	String	変更後の出力設定のファイル名

CreateAgtByString

パス名からエージェントを生成する

[形式]

CreateAgtByString (arg1)

[引数]

引数名	型	説明
arg1	String	生成するエージェントのパス名

[戻り値]

型	説明
Agt	エージェント変数

CreateAgtMulti

複数のエージェントを生成する

[形式]

CreateAgtMulti (arg1, arg2)

[引数]

引数名	型	説明
arg1	AgtType	エージェント型変数
arg2	Integer	生成するエージェント数

[戻り値]

型	説明
AgtSet	生成したエージェント集合

DefineAgtType

エージェント型を定義する

[形式]

DefineAgtType(arg1, arg2, arg3, arg4)

[引数]

引数名	型	説明
arg1	String	追加先のパス名
arg2	String	追加するエージェント型の名称
arg3	String	追加するエージェント変数のファイル名
arg4	String	追加するエージェントルールのファイル名

DefineSpace

空間を定義する

[形式]

DefineSpace(arg1, arg2, arg3, arg4, arg5, arg6, arg7, arg8)

[引数]

引数名	型	説明
arg1	String	追加先のパス名
arg2	String	追加する空間の空間名
arg3	String	追加する空間の空間種別 (格子モデルのときはSquare_2D、六角モデルのときはHexagon_2D)
arg4	Integer	追加する空間の大きさX
arg5	Integer	追加する空間の大きさY
arg6	String	追加する空間のループ種別 (ループするときはLoop、しないときは!Loop)
arg7	String	原点位置 (必ず "North" と入力)
arg8	Integer	レイヤ数

DefineVariable

新規に変数を定義する

[形式]

DefineVariable(arg1, arg2)

[引数]

引数名	型	説明
arg1	String	追加先のパス名
arg2	String	追加する変数のファイル名

SaveModel

モデルファイルを保存する (※Professional Only)

[形式]

SaveModel (arg1)

[引数]

引数名	型	説明
arg1	String	保存するモデルファイル名

SpaceToPath

空間変数の値からパス名を取得する

[形式]

SpaceToPath (arg1)

[引数]

引数名	型	説明
arg1	Space	空間変数

[戻り値]

型	説明
String	パス名

TranscriptValue

エージェントが保持する変数の値を他のエージェントに転写する

[形式]

TranscriptValue (arg1, arg2)

[引数]

引数名	型	説明
arg1	Agt	転写先のエージェント
arg2	Agt	転写元のエージェント

ユーザ定義

名称	説明
Function	返り値を戻すユーザ定義関数。ユーザ定義関数の実行時に、その結果の値 (Return 値) を返すときに使用する
Sub	返り値のないユーザ定義関数。ユーザ定義関数の実行時に、その結果の値を返さないときに使用する

Function

返り値を戻すユーザ定義関数。ユーザ定義関数の実行時に、その結果の値 (Return 値) を返すときに使用する

[形式]

```
Function ユーザ定義関数名 (パラメータ宣言) As 関数戻り値のデータ型
{
    変数の型宣言部
    実行部
    Return (式)
}
```

Sub

返り値のないユーザ定義関数。ユーザ定義関数の実行時に、その結果の値を返さないときに使用する

[形式]

```
Sub 関数名 (パラメータ宣言)
{
    変数の型宣言部
    実行部
}
```

名称	説明
ClearConsoleScreen	コンソール画面をクリアする
ClearDebugScreen	デバッグ画面をクリアする ※デバッグ画面内でのみ利用可能
EvalPnuts	ExecPnutsで定義した関数を呼び出す (※Professional Only)
EvalScript	スクリプト文字列を実行する (※Professional Only)
ExecPnuts	Pnutsスクリプトを実行する ※Univ_Initで実行可能 (※Professional Only)
ExecScript	スクリプトファイルを実行する (※Professional Only)
ExitSimulation	シミュレーションを終了する。関数実行後、Univ_Finishを実行してシミュレーションを終了する
ExitSimulationMsg	文字列をメッセージに出力し、シミュレーションを終了する。関数実行後、Univ_Finishを実行してシミュレーションを終了する
ExitSimulationMsgLn	文字列をメッセージに出力し(改行つき)、シミュレーションを終了する。関数実行後、Univ_Finishを実行してシミュレーションを終了する
GetArraySize	指定された配列変数のサイズを取得する
GetClickedMapPosition	マップ出力画面でクリックしたマップ上のX-Y座標を返す。※この関数は、IsMouseClickedOnMapがTrueを返すときのみ利用可能
GetCountSimulationNumber	現在のシミュレーション実行回数を取得する
GetCountStep	現在のステップ数を取得する
GetIPAddress	IPアドレスを取得する (※Professional Only)
GetLastErrorMsg	実行時に発生したエラーを取得する
Gradation	グラデーション色を取得する
IsMouseClickedOnMap	マップ出力画面でマウスクリックが行われたか否かを返す
Print	改行なしでコンソール画面に出力する
PrintLn	改行つきでコンソール画面に出力する
RGB	RGB値を返す
ScreenShot	最前面の出力画面のスクリーンショットを撮り、JPEG画像ファイルとして保存する
ScreenShotJPG	指定した出力画面(マップ出力, グラフ出力のみ)のスクリーンショットを撮り、JPEG画像ファイルとして保存する
ScreenShotPNG	指定した出力画面(マップ出力, グラフ出力のみ)のスクリーンショットを撮り、PNG画像ファイルとして保存する
Shell	実行可能な外部プログラムを実行する

ClearConsoleScreen

コンソール画面をクリアする

[形式]

ClearConsoleScreen ()

ClearDebugScreen

デバッグ画面をクリアする ※デバッグ画面内でのみ利用可能

[形式]

ClearDebugScreen()

EvalPnuts

ExecPnutsで定義した関数を呼び出す (※Professional Only)

[形式]

EvalPnuts (arg1)

[引数]

引数名	型	説明
arg1	String	ExecPnutsで定義した関数名

[戻り値]

型	説明
String	関数を評価した結果の文字列

EvalScript

スクリプト文字列を実行する (※Professional Only)

[形式]

EvalScript (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	スクリプト言語名 (「ruby」が指定可能)
arg2	String	スクリプト文字列

[戻り値]

型	説明
String	実行結果の文字列

ExecPnuts

Pnutsスクリプトを実行する ※Univ_Initで実行可能 (※Professional Only)

[形式]

ExecPnuts (arg1)

[引数]

引数名	型	説明
arg1	String	Pnutsスクリプトのモデルファイルからの相対パス

[戻り値]

型	説明
Integer	エラー値（正常のときは0、エラーのときは-1）

ExecScript

スクリプトファイルを実行する（※Professional Only）

[形式]

ExecScript (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	スクリプト言語名（「ruby」が指定可能）
arg2	String	スクリプトファイルのモデルファイルからの相対パス

ExitSimulation

シミュレーションを終了する。関数実行後、Univ_Finishを実行してシミュレーションを終了する

[形式]

ExitSimulation()

ExitSimulationMsg

文字列をメッセージに出力し、シミュレーションを終了する。関数実行後、Univ_Finishを実行してシミュレーションを終了する

[形式]

ExitSimulationMsg (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long, String	出力したい文字列

ExitSimulationMsgLn

文字列をメッセージに出力し（改行つき）、シミュレーションを終了する。関数実行後、Univ_Finishを実行してシミュレーションを終了する

[形式]

ExitSimulationMsgLn (arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long, String	出力したい文字列

GetArraySize

指定された配列変数のサイズを取得する

[形式]

GetArraySize (arg1)

[引数]

引数名	型	説明
arg1	Boolean(), Double(), Integer(), Long(), String(), Agt()	配列変数

[戻り値]

型	説明
Integer	配列数

GetClickedMapPosition

マップ出力画面でクリックしたマップ上のX-Y座標を返す。※この関数は、IsMouseClickedOnMapがTrueを返すときのみ利用可能

[形式]

GetClickedMapPosition (arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Double	X座標
arg2	Double	Y座標
arg3	String	マップ名

GetCountSimulationNumber

現在のシミュレーション実行回数を取得する

[形式]

GetCountSimulationNumber ()

[戻り値]

型	説明
Integer	シミュレーション実行回数

GetCountStep

現在のステップ数を取得する

[形式]

GetCountStep()

[戻り値]

型	説明
Long	ステップ数

GetIPAddress

IPアドレスを取得する (※Professional Only)

[形式]

GetIPAddress()

[戻り値]

型	説明
String	IPアドレス

GetLastErrorMsg

実行時に発生したエラーを取得する

[形式]

GetLastErrorMsg()

[戻り値]

型	説明
String	エラーメッセージ

Gradation

グラデーション色を取得する

[形式]

Gradation(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Integer	開始色
arg2	Integer	終了色
arg3	Double	度合い (0 to 1で指定)

[戻り値]

型	説明
Integer	グラデーション色 (引数が不正なときは-1)

IsMouseClickedOnMap

マップ出力画面でマウスクリックが行われたか否かを返す

[形式]

IsMouseClickedOnMap()

[戻り値]

型	説明
Boolean	判定値 (マウスをクリックしたときはTrue、していないときはFalse)

Print

改行なしでコンソール画面に出力する

[形式]

Print(arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long, String	出力したい文字列

PrintLn

改行つきでコンソール画面に出力する

[形式]

PrintLn(arg1)

[引数]

引数名	型	説明
arg1	Double, Integer, Long, String	出力したい文字列

RGB

RGB値を返す

[形式]

RGB(arg1, arg2, arg3)

[引数]

引数名	型	説明
arg1	Integer	R値 (0 to 255)
arg2	Integer	G値 (0 to 255)
arg3	Integer	B値 (0 to 255)

[戻り値]

型	説明
Integer	RGB値 (R値、G値、B値のいずれかが無効な値のときは-1)

ScreenShot

最前面の出力画面のスクリーンショットを撮り、JPEG画像ファイルとして保存する

[形式]

ScreenShot (arg1)

[引数]

引数名	型	説明
arg1	String	JPEG画像ファイルの出力パス名

ScreenShotJPG

指定した出力画面(マップ出力, グラフ出力のみ)のスクリーンショットを撮り、JPEG画像ファイルとして保存する

[形式]

ScreenShotJPG (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	マップおよびグラフ出力画面名(マップ名, グラフ名)
arg2	String	JPEG画像ファイルの出力パス名

ScreenShotPNG

指定した出力画面(マップ出力, グラフ出力のみ)のスクリーンショットを撮り、PNG画像ファイルとして保存する

[形式]

ScreenShotJPG (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	マップおよびグラフ出力画面名(マップ名, グラフ名)
arg2	String	PNG画像ファイルの出力パス名

Shell

実行可能な外部プログラムを実行する

[形式]

Shell (arg1, arg2)

[引数]

引数名	型	説明
arg1	String	実行可能な外部プログラムのパス名
arg2	Boolean	実行モード(同期実行するときはTrue、非同期実行のときはFalse)

定数

名称	説明
COLOR_BLACK	黒色を表す
COLOR_BLUE	青色を表す
COLOR_CYAN	水色を表す
COLOR_GREEN	緑色を表す
COLOR_MAGENTA	紫色を表す
COLOR_RED	赤色を表す
COLOR_WHITE	白色を表す
COLOR_YELLOW	黄色を表す
FILE_APPEND	追記モードを表す
FILE_READ	読み込みモードを表す
FILE_WRITE	上書きモードを表す
False	ブール型の値で偽を表す
True	ブール型の値で真を表す

COLOR_BLACK

黒色を表す

[形式]

COLOR_BLUE

青色を表す

[形式]

COLOR_CYAN

水色を表す

[形式]

COLOR_GREEN

緑色を表す

[形式]

COLOR_MAGENTA

紫色を表す

[形式]

COLOR_RED

赤色を表す

[形式]

COLOR_WHITE

白色を表す

[形式]

COLOR_YELLOW

黄色を表す

[形式]

FILE_APPEND

追記モードを表す

[形式]

FILE_READ

読み込みモードを表す

[形式]

FILE_WRITE

上書きモードを表す

[形式]

False

ブール型の値で偽を表す

[形式]

True

ブール型の値で真を表す

[形式]

演算子

名称	説明
And	かつ(論理積)
Mod	整数の除算の余り(剰余)(小数点以下の値は切り捨て)
Not	ではない(否定)
Or	または(論理和)
Xor	どちらかが真(排他的論理和)

And

かつ(論理積)

[形式]

A And B (AかつB)

Mod

整数の除算の余り(剰余)(小数点以下の値は切り捨て)

[形式]

one = 5 Mod 3 (oneに 2を代入)
two = 7 Mod 3 (twoに 1を代入)

Not

ではない(否定)

[形式]

Or

または(論理和)

[形式]

A Or B (AまたはB)

Xor

どちらかが真(排他的論理和)

[形式]

A Xor B (AまたはBのどちらかが真)

名称	説明
Break	飛び越し文で使用する
Do	Do While文、Do Until文で使用する
Each	For文で使用する
Else	If文で使用する
Elseif	If文で使用する
End	If文で使用する
For	For文で使用する
Goto	Goto文で使用する
If	If文で使用する
In	For文で使用する
Loop	Do While文、Do Until文で使用する
Next	For文で使用する
Step	For文で使用する
Then	If文で使用する
To	For文で使用する
Until	Do Until文で使用する
While	Do While文で使用する

Break

飛び越し文で使用する

[形式]

Do

Do While文、Do Until文で使用する

[形式]

Each

For文で使用する

[形式]

Else

If文で使用する

[形式]

ElseIf

If文で使用する

[形式]

End

If文で使用する

[形式]

For

For文で使用する

[形式]

Goto

Goto文で使用する

[形式]

If

If文で使用する

[形式]

In

For文で使用する

[形式]

Loop

Do While文、Do Until文で使用する

[形式]

Next

For文で使用する

[形式]

Step

For文で使用する

[形式]

Then

If文で使用する

[形式]

To

For文で使用する

[形式]

Until

Do Until文で使用する

[形式]

While

Do While文で使用する

[形式]

名称	説明
Agt_Init	シミュレーションで最初に1回だけ実行される関数の名称
Agt_Step	シミュレーションで繰り返し実行される関数の名称
All	※互換性のために予約
As	宣言時に使用する
By	※将来のために予約
Dim	宣言時に使用する
My	当該エージェントを表す
Univ_Finish	シミュレーションが終了する直前に1回実行される関数の名称
Univ_Init	シミュレーションが開始された最初に1回実行される関数の名称
Univ_Step_Begin	シミュレーションのステップの最初に実行される関数の名称
Univ_Step_End	シミュレーションのステップの最初に実行される関数の名称
Universe	Universeを表す
Up	※互換性のために予約
Val	※将来のために予約
World	※互換性のために予約

Agt_Init

シミュレーションで最初に1回だけ実行される関数の名称

[形式]

```
Agt_Init{
    最初に1回だけ実行させるルール
}
```

Agt_Step

シミュレーションで繰り返し実行される関数の名称

[形式]

```
Agt_Step{
    毎ステップ実行させるルール
}
```

All

※互換性のために予約

[形式]

As

宣言時に使用する

[形式]

By

※将来のために予約

[形式]

Dim

宣言時に使用する

[形式]

My

当該エージェントを表す

[形式]

Univ_Finish

シミュレーションが終了する直前に1回実行される関数の名称

[形式]

```
Univ_Finish{  
    最後に1回だけ実行させるルール  
}
```

Univ_Init

シミュレーションが開始された最初に1回実行される関数の名称

[形式]

```
Univ_Init{  
    最初に1回だけ実行させるルール  
}
```

Univ_Step_Begin

シミュレーションのステップの最初に実行される関数の名称

[形式]

```
Univ_Step_Begin{  
    毎ステップの最初に実行させるルール  
}
```

Univ_Step_End

シミュレーションのステップの最初に実行される関数の名称

[形式]

```
Univ_Step_End{  
    毎ステップの最後に実行させるルール  
}
```

Universe

Universeを表す

[形式]

Up

※互換性のために予約

[形式]

Val

※将来のために予約

[形式]

World

※互換性のために予約

[形式]

名称	説明
Static	※将来のために予約
Variant	※将来のために予約

Static

※将来のために予約

[形式]

Variant

※将来のために予約

[形式]