

● list モジュール述語

listモジュールはシステムに組み込まれた標準のライブラリモジュールの集合です。
使用する前には以下のように、インクルードしてから使います。

```
? <include list>;
```

呼び出すときには、`::list`に続けて記述します。

```
::list <append #out #in1 #in2>
```

#in1と#in2のリストを連結して、#outに設定します。

```
::list <reverse #out #in>
```

リスト#inを逆の順に変換して#outに設定します。

```
::list <member #mem #list>
```

#listの中に#memが含まれているか判定します。

```
::list <last #out #list>
```

リスト#listの最後の要素を、#outに設定します。

```
::list <flatten #out #in>
```

リスト#inの中の入れ子状のリストを、平坦なリストに変換して#outに設定します。

```
::list <difference #out #in1 #in2>
```

リスト#in1と#in2のリストの要素の中の共通でない要素を差分として、#in1にあるが#in2にない要素のリストを#outに設定します。

```
::list <intersect #out #in1 #in2>
```

リスト#in1と#in2の要素を比較し、共通な要素のリストを#outに設定します。要素の順番に関わらず、共通の要素が抽出されます。

```
::list <union #out #in1 #in2>
```

リスト#in1と#in2の要素を併せ、共通する要素は重複しないように一つにして、すべての要素を#outに設定します。

```
::list <subset #subset #set>
```

#subsetが#setのサブセットであるときtrueになります。

```
::list <join #out #in1 #in2>
```

#in1と#in2を併合したリストを#outに設定します。併合は#in1と#in2の対応する順番の要素ごとに行われます。つまり、#in1の1番目と#in2の

1 番目が#outに追加され、次は 2 番目同士の要素が並び、これをリストの最後の要素まで繰り返して#outに設定していきます。

```
::list <occures #n #e #list>
```

要素#eが、#listの中で出現する回数を、#nに設定します。