

XMP/Cの指示文中における角括弧

- ノード集合やテンプレートにおいて、C言語のユーザにとってFortranの並び順と1-originであることは慣れていないので、XMP/Cにおいては角括弧を用いることで、C言語の並び順と0-originも利用できるようにする

現仕様

```
#pragma xmp nodes p(2, 4)
#pragma xmp template t(0:M-1, 0:N-1)
#pragma xmp distribute t(block, cyclic(2)) onto p
int a[N][M];
#pragma xmp align a[i][j] with t(j, i)

#pragma xmp loop (j, i) on t(j, i)
for( i = 0; i < N; i++)
  for( j = 0; j < M; j++)
    a[i][j] = func(i, j);
```

提案仕様

```
#pragma xmp nodes p[4][2]
#pragma xmp template t[N][M]
#pragma xmp distribute t[cyclic(2)][block] onto p
int a[N][M];
#pragma xmp align a[i][j] with t[i][j]

#pragma xmp loop (i,j) on t[i][j]
for( i = 0; i < N; i++)
  for( j = 0; j < M; j++)
    a[i][j] = func(i, j);
```

角括弧はCの並び順かつ0-origin

他の例

現仕様

```
#pragma xmp nodes p(16)
#pragma xmp nodes q(4,*)
#pragma xmp nodes r(8) = p(3:10)
#pragma xmp nodes z(2,3) = p(1:6)
```

提案仕様

```
#pragma xmp nodes p[16]
#pragma xmp nodes q[*][4]
#pragma xmp nodes r[8] = p[3:8]
#pragma xmp nodes z[3][2] = p[0:6]
```

特徴：

- Cのtripletの適用 [start:length]
- アスタリスクは最初の次元のみ利用可能

- 影響を受ける指示文と関数
 - nodes指示文およびtemplate指示文
 - 他の指示文は、角括弧を用いて表現したノード集合やテンプレートをon節などで参照する際に影響を受ける
 - xmp_node_num(), xmp_all_node_num(), xmp_nodes_index()で得る値は0-origin

Coarrayについて

現仕様

```
int a[100]:[2][*];  
  
if(xmp_node_num()==1)  
  a[:,2][5] = a[:,];
```

提案仕様

```
int a[100]:[*][2];  
  
if(xmp_node_num()==0)  
  a[:,4][1] = a[:,];
```

nodes指示文とxmp_node_num()に対応させるため、XMP/Cではimage集合も0-originかつC言語の並び順にしたい