Subject: Re: Set Operations on A and B Posted by btt on Mon, 10 Oct 2005 13:46:16 GMT

View Forum Message <> Reply to Message

mxd1007@cs.rit.edu wrote:

```
>
  setA = \{vertex0: [1,5,8], vertex1: [0,2,7], vertex2: [,3,6,7\}, \}
       vertex3:[2,4,10],vertex4:[3,5,10],vertex5:[0,4,6,8],$
>
       vertex6:[2,5,9],vertex7:[1,2,9,10],vertex8:[0,5,9,10],$
>
       vertex9:[6,7,8,10],vertex10:[3,4,7,8,9]}
>
snip
>
> ;retrieve subarray from structure setA
  newSetA = setA.(i)
  sizeColorClass = SIZE(colorClass[h], /n_dimensions)
  WHILE( h LT k AND SetIntersection(newSetA, REFORM(colorClass[h],$
       sizeColorClass) NE -1) DO......
>
>
```

Two things, I think, will help here.

(1) You probably want change the call to size so that it returns a vector of dimensions - not a scalar of the number of dimensions. I think your REFORM statement will work if you do since it will work with

sizeColorClass = SIZE(colorClass[h], /dimensions)

(2) I would seriously be looking transforming this whole thing into an object system. Make each vertex an object. Then you have two options - have each vertex know something about its adjacency or have another type of object provide supervision/navigation for you. I did something similar with a quadtree object system a long time ago. Once I had my head wrapped around the object part - the quadtree navigation was a snap. And a bonus was waaaay less coding than that without objects.

Cheers, Ben

Hello,