Subject: Re: Set Operations on A and B
Posted by mxd1007 on Sat, 08 Oct 2005 01:22:08 GMT
View Forum Message <> Reply to Message

Ben Tupper wrote:
> mxd1007@cs.rit.edu wrote:
>
>> which refers to Colorclass[h]. So I guess since this is only a
>> numerical value, I need to somehow convert it to an array.
>> ColorClass[h] is the set of vertices with color h, and is constructed
>> as follows:
>>
>> ColorClass[h] = {i is an element of the set of vertices : color[i] = h}
>> for 0 <= h <= k-1
>>
>
> Hi,
>
> My first guess is that IDL is changing your selection from a subset of
> an array to a scalar. This happens when you extract one element from an
> array. Use REFORM to force IDL to treat your value as an array of one
> element.
>
> IDL> ColorClass = BINDGEN(12)
> IDL> i = 7
>
> Now extract the ith element - raw and reformed...
>
> IDL> help, ColorClass[i], REFORM(ColorClass[i],1)
> <Expression> BYTE = 7
> <Expression> BYTE = Array[1]
>
> Try it with your set operators.
>
> Cheers,
> Ben

Thanks Ben

I changed the set of adjacent vertices to a larger graph, 11 vertices,
and the structure of the graph is as follows, where i is the vertex and
A[i] is the set of vertices adjacent to vertex i, however A[i] had to
be a struct instead of an array

```
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]}

knowing that this graph can have at least 3 different colors on vertices(a color must not be connected to the same color on any vertex, so vertex 0 would be green, vertex 1 would be red, vertex 2 would be blue, any adjacent vertices have to be of different colors) so I constructed an array called colorClass,

```
 colorclass=make_array(3,1, value=-1)
```

where initially this array has to be NULL or empty.

so to pass in colorClass[h] into the setIntersection function, I used the reform function,

```
k=0
FOR i = 0, 10 DO BEGIN
h = 0

;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.....
```

my question is, colorClass[h], h = 0,1,.....10 is going to change in size, perhaps colorclass[0] = {0,2,4}, colorClass[1] = {1,3,5,8}, colorClass[2] = {6,7,9,10} but the highest h will be is 2, so with that, am I passing in the correct numeric value for the reform fucntion for colorClass? I keep getting this error and trying to figure out how to pass colorClass[h] to my setIntersection function where colorClass[h] could be an empty set, a set with 1 elements, or 2 elements, or 3 elements, etc etc...

REFORM: New subscripts must not change the number elements in <INT Array[1]>.
