Subject: Re: Unpacking algorithm
Posted by Tmorri on Sat, 15 May 2004 23:00:19 GMT
View Forum Message <> Reply to Message

one second thought,

Does any one have an algorithm to unpack this vector

v0=[0 x1 x2 x3 0 x4 x5 x1 0 x2 x3 x4 0 x5 x1 x2 0 x3 x4 x5]

in the following way:

v1=[0 0 0 0 0]

v2=[x1 x2 x3 x4 x5 x1 x2 x3 x4 x5 x1 x2 x3 x4 x5]

x1,x2,x3,x4,x5 are variables that can take any value, even zero, (0).

I just want to get rid othe zeroes (every fourth element)shown in vector
v0

Thanks,

Tmorri

Subject: Re: Unpacking algorithm
Posted by Christopher Lee on Sun, 16 May 2004 15:19:03 GMT
View Forum Message <> Reply to Message

> v0=[0 x1 x2 x3 0 x4 x5 x1 0 x2 x3 x4 0 x5 x1 x2 0 x3 x4 x5]
>
> I just want to get rid othe zeroes (every fourth element)shown in vector
> v0

```
n=n_elements(v0)
s=4
index=(findgen(s,fix((n+3)/s)))[1:s-1,*]
index=reform(index, n_elements(index))
;index skips all of the mod 4 indices
new_v=v0[index]

;or even shorter, if n_elements(v0) mod s = 0

s=4
```

```
 v0=reform((reform(v0,s,n_elements(v0)/s))[1:s-1,*],n_element s(v0)*(s-1)/s)
```

Chris.

Subject: Re: Unpacking algorithm
Posted by MKatz843 on Mon, 17 May 2004 04:37:17 GMT

View Forum Message <> Reply to Message

You could also use where() to find the indices of the "zero" elements,
or their complement, that is the indices of the non-zero elements.

```
v0=[0 x1 x2 x3 0 x4 x5 x1 0 x2 x3 x4 0 x5 x1 x2 0 x3 x4 x5]
w2 = where(v0 NE 0, complement=w1)  ;--- returns an array of indices
v2 = v0(w2)
v1 = v0(w1)
```

There are also very handy keywords: count and ncomplement that return
the number of indices of each kind that are found. Note that where()
returns -1 when there are no matches. If you try to use v0(w1) and w1
is -1 you'll get an error. That's where the count variables come in. I
usually do something like "if count GT 0 then (do something)" to avoid
the error.

M. Katz

"Tmorri" <torrimorri@yahoo.com> wrote in message
news:<52e0072063f360e0705e8ac96f274e52@localhost.talkaboutprogramming.com>...

> one second thought,
>
> Does any one have an algorithm to unpack this vector
>
>
> v0=[0 x1 x2 x3 0 x4 x5 x1 0 x2 x3 x4 0 x5 x1 x2 0 x3 x4 x5]
>
> in the following way:
>
> v1=[0 0 0 0 0]
>
> v2=[x1 x2 x3 x4 x5 x1 x2 x3 x4 x5 x1 x2 x3 x4 x5]
>
> x1,x2,x3,x4,x5 are variables that can take any value, even zero, (0).
>
> I just want to get rid othe zeroes (every fourth element)shown in vector
> v0
>
>

> Thanks,
>
> Tmorri

Subject: Re: Unpacking algorithm
Posted by Timm Weitkamp on Mon, 17 May 2004 13:41:50 GMT
View Forum Message <> Reply to Message

Tmorri:

If I have understood your problem, then the following might do the job
(given your vector v0, of which you want to get every 4th element into
v1 and the others into v2).

```
 i0 = INDGEN(N_ELEMENTS(v0))
 i1 = WHERE((i0 MOD 4) EQ 0, COMPLEMENT=i2)
 v1 = v0[i1]
 v2 = v0[i2]
```

Hope this helps,
Timm

PS. I'm sure there's a better way than using WHERE, but I don't like to
think.

On 15.05.04 at 19:00 -0400, Tmorri wrote:

> one second thought,
>
> Does any one have an algorithm to unpack this vector
>
>
> v0=[0 x1 x2 x3 0 x4 x5 x1 0 x2 x3 x4 0 x5 x1 x2 0 x3 x4 x5]
>
> in the following way:
>
> v1=[0 0 0 0 0]
>
> v2=[x1 x2 x3 x4 x5 x1 x2 x3 x4 x5 x1 x2 x3 x4 x5]
>
> x1,x2,x3,x4,x5 are variables that can take any value, even zero, (0).
>
> I just want to get rid othe zeroes (every fourth element)shown in vector
> v0
>
>
> Thanks,

>
> Tmorri
>
>