Thanks David and Craig. If Craig says "no", this means something.
Although I'd wait with the verdict until I hear from JD :-) After all,
the loop I have with Total, although it goes through all elements of B,
is taking only 0.03 s on my machine for B with ~2500 points converted to
C with ~50k points, which is acceptable since it is not executed many
times repeatedly. In contrast with looping using Replicate that was
taking 5.5 s :-(

Cheers,
Pavel

"Pavel A. Romashkin" wrote:
>
> If I have
>
> a = findgen(10)
> b = fix(100* randomu(10, 10))
> ; N_elements(a) is equal to n_elements(b)
> c = findgen(total(b))
>
> how can I fill C with values from A using B as a running index, so that
>
> c[0 : b[0]-1] = a[0]
> c[b[0] : b[0]+b[1]-1] = a[1]
>
> etc, without looping through "n_elements(b)-1" iterations?
> I have a fast solution with a loop and indexing using total(/cumulative)
> and a very slow one with loop and replicate, but I can't come up with a
> loop-free one.
>
> Thank you,
> Pavel