>

Simulation of the ice-cream ordering problem from Physics Forums. The five types of ice creams are {1,2,3,4,5}, of which we choose 3 at random (without replacement), and are interested in B = number of samples containing ice-cream '1'. There are four toppings {1,2,3,4}, of which we choose 2 and are interested in C = number of samples containing topping '1'. We are also interested in BC = number of samples containing both ice-cream '1' and topping '1'.

Below, we use the Maple package 'randcomb(n,k)' to pick a random subset of size k from {1,2,...,n}, then test if 1 is in the subset. We do that my naming the subset, then doing a Boolean evaluation (evalb ()) to see if the statement "1 in the subset" is true or false. We then convert true/false ot 1/0 for purposes of tallying up the totals. We do the same type of operations for choosing the toppings.

With each draw we leave B unchanged (if the 'flavor' subset does not contain '1') or increment it by 1 otherwise. Similarly for the toppings total C. Finally, we increment BC by 1 if and only if both B and C were incremented.

```
> restart;
> with(Statistics) :
> with(combinat);
```

[Chi, *bell*, binomial, *cartprod*, *character*, *choose*, *composition*, *conjpart*, *decodepart*, *encodepart*, *eulerian1*, *eulerian2*, *fibonacci*, *firstcomb*, *firstpart*, *firstperm*, *graycode*, *inttovec*, *lastcomb*, *lastpart*, *lastperm*, *multinomial*, *nextcomb*, *nextpart*, *nextperm*, *numbcomb*, *numbcomp*, *numbpart*, *numbperm*, *partition*, *permute*, *powerset*, *prevcomb*, *prevpart*, *prevperm*, *randcomb*, *randpart*, *randperm*, *rankcomb*, *rankperm*, *setpartition*, stirling1, stirling2, *subsets*, *unrankcomb*, *unrankperm*, *vectoint*] **(1)**

```
> randomize( ) : # <--- initialize the random number generator
> B := 0; C := 0; BC := 0; # <-- initial totals
```

$$B := 0$$
$$C := 0$$
$$BC := 0 \quad \textbf{(2)}$$

```
>
> N := 10000; # <-- number of samples
```

$$N := 10000 \quad \textbf{(3)}$$

```
> for i from 1 to N do
    xb := 1 in randcomb(5, 3) :
    Ib := subs( {false = 0, true = 1}, evalb(xb) ) :
    xc := 1 in randcomb(4, 2) :
    Ic := subs( {false = 0, true = 1}, evalb(xc) ) :
   B := B + Ib : C := C + Ic : BC := BC + Ib·Ic :
   end do:

> B, C, BC;
```

$$6075, 4980, 3027 \quad \textbf{(4)}$$

So, in this sample of 10,000 ice-cream cone orders:
(1) flavor '1' occurred B = 6075 times (so the sample probability is 0.6075, quite close to the theoretical 60%);

(2) the topping '1' occurred C = 4980 times (so the sample probability is 0.4980, quite close to the theoretical 50%);
(3) the flavor-topping combination '1 1' occurred BC = 3027 times (so the sample probability is 0.3027, quite close to
the theoretical 30%)

>