

```

> IntMul := proc( m::posint, a::Array,
                  n::posint, b::Array,
                  c::Array, # space for product
                  B::posint )
  local i,j,t,carry;
  for i from 0 to m+n-1 do
    c[i] := 0;
  od;
  for i from 0 to m-1 do
    carry := 0;
    for j from 0 to n-1 do

      t := b[i]*a[j] + c[i+j] + carry;
      c[i+j] := irem(t,B,'carry');

    od;

    c[i+j] := carry;
  od;
  # return the length of the product
  if carry=0 then n+m-1 else n+m fi;

end:

> a := Array(0..2,[5,4,3]): # a = 345
> b := Array(0..2,[6,7,8]): # b = 876
> c := Array(0..5):
> n := IntMul(3,a,3,b,c,10);
                                     n := 6
> 345*876 = add( c[i]*10^i, i=0..n-1 );
                                     302220 = 302220

```