```
****************************  FP HELP SHEET  *****************************  C. Bays
There is much information here, but you need to study the material closely.
TO PRINT THIS OUT ON ONE PAGE: (1) GO TO I.E.  (2) View->Text Size->Smallest (3) Print
  tl:<1 2 3> => <2 3>        (tail)
   +:<5 6>    =>   11        (add)  (other simple functions are introduced throughout)
******** three "functional forms" :   [ ],  @,  and %  *******
  [+,-,*,/]:<2 3>       => <5 -1 6 0.666666667>      (construction)
  tl @ tl:<1 2 3 4 5 6>  =>  <3 4 5 6>   (composition)
  %35:<1 2 3 4 5 6 7>   => 35   (constant)
  %<2 <4 5>>: F  =>  <2 <4 5> >  (note: F and T are valid values, as is, eg., , etc.)
*********** more functions *******
  apndl:< 8 <2 3 4>>   =>     <8 2 3 4>
  rotr:<1 2 3 4>    =>     <4 1 2 3>
  1:<6 7 8 9 10>    =>     6      (selection)
  id:<3 4 5 < 7 8>>   =>    <3 4 5 < 7 8>>  (identity)
******** two important functional forms:  &  and ! ( apply-to-all  and  insert ) ******
  &+:<<2 3><4 5>>    =>    <5 9>  (apply to all  add)
  insert defined: !F:< x1 x2 . . xn > =F:< x1 !F:< x2 . . xn >> . . etc; and !F:< xn > = xn
  !+:<2 3 4 5>   =>    14     (insert add)
  !*:<2 3 4 5>   =>    120     (insert multiply)
  !id:<2 3 4 5>  => <2 <3 <4 5>>>
  !tl:<1 2 3 4 5> => <<<<5> > > >  (insert can produce some unusual results.)
  + @ &!+:<<1 2 3 4 5><6 7 8 9 10>>  =>  55  (apply-to-all insert add, then add the pair)
  [+@&+,id]:<<2 3><4 5>>    =>    <14 <<2 3> <4 5> > >  (construction and identity)
  &[id,id]:< 2 3 4 5 6> => <<2 2><3 3><4 4><5 5><6 6>> (apply-to-all construct.See below)
  [id,id]:< 2 3 4 5 6>    =>    <<2 3 4 5 6><2 3 4 5 6>>
  concat:<<1 23 ><4 5 6><6 7 8>>    =>    <1 23 4 5 6 6 7 8>
  concat@[ [1], tl ]:<1 2 3 4 5>    =>    <1 2 3 4 5>
  =:<1 2> =>  F  (boolean operation '=' )  also there are '>' (greater than), '<' ,  etc.
  =:<3 3> =>  T
  =:<<2 3 4> < 2 3 4>>    =>  T
************** if and while ***************
  ( condition -> doThisIfConditionTrue ; doThisIfConditionFalse )
  (=@[1,%7] ->tl ; id):<4 7 8 9> => <4 7 8 9> (if first element =7 delete; else do nothing)
  (=@[1,%7] ->tl ; id):<7 4 8 9> => <4 8 9> ( if first element =7 delete; else do nothing)
  (while conditionIsTrue ; continueToDoThis)
  (while =@[1,%0] ; tl):<0 0 0 0 0 0 2 3 6>   =>    <2 3 6>  ( while the first element  = 0
      continue to take the tail;  i.e. strip off leading zeros)
**************creating a function *****************
  {abs (<@[id,%0] -> -@[%0,id]  ;  id) }
{abs} ( note fp prompt at left.  The "abs"  above:  if the element < 0 then make negative)
  abs:3    =>    3
  abs:-6    =>    6
  &abs:<2 3 4 -5 -6 -2 7 8 9 -2>    =>    <2 3 4 5 6 2 7 8 9 2>
  {fact (=@[1,%1] -> %1 ; * @ [ 1,  fact @ [ - @ [1,%1] ] ] ) }
{fact}
  fact:<3>    =>    6
  fact:<5>    =>    120
***************** array operations ***************
  trans:<<1 2 3><4 5 6><7 8 9><101 102 103>>     =>
<<1 4 7 101> <2 5 8 102><3 6 9 103>>
  distl:<<2 1 3> <<1 2 3><4 5 6><7 8 9><101 102 103>>>    =>
<<<2 1 3><1 2 3>><<2 1 3><4 5 6>><<2 1 3><7 8 9>><<2 1 3><101 102 103>>>
  (distribute left.  also,  distr)
  &trans@ distl:<<2 1 3> <<1 2 3><4 5 6><7 8 9><101 102 103>>>    =>
<<<2 1><1 2><3 3>> <<2 4><1 5><3 6>><<2 7><1 8><3 9>><<2 101><1 102><3 103>>>
  &&*@ &trans@ distl:<<2 1 3><<1 2 3><4 5 6><7 8 9><101 102 103>>>    =>
<<2 2 9><8 5 18><14 8 27><202 102 309>>
  &!+@ &&*@ &trans@ distl:<<2 1 3><<1 2 3><4 5 6><7 8 9><101 102 103>>>    =>
<13 31 49 613>
 (the above is vector - matrix multiplication)
(now,  here is vector - vector multiplication (also called "inner product"))
  trans:<<1 2 3 4 5><6 7 8 9 10>>    =>
<<1 6><2 7><3 8><4 9><5 10>>
```

```
  &*@ trans:<<1 2 3 4 5><6 7 8 9 10>>   =>    <6 14 24 36 50>
  !+@ &*@ trans:<<1 2 3 4 5><6 7 8 9 10>>   =>    130
now, create the function
  {ip  !+@ &*@ trans}
{ip}
  ip:<<1 2 3 4 5><6 7 8 9 10>>   =>    130
```