* 1st program (spmm1): use matrix multiplication algorithm
* 2nd program (spfw1): Floyd-Warshall algorithm
* all 2 programs allow the user to input a source/sink node repeatedly until the user enters a terminal letter(i.e. ‘n’)

example1: suppose you execute **spmm1**, you should see:

**Please input network filename:**(then you input, say, input\_100\_400\_1.sp)  
**Please input a source node** **[input 'n' to stop]:** (then you input, say, 2)

**Please input a sink node:** (then you input, say, 20)

then you should see the output like

**2->20: [9]**

if you input source node as node 10, sink node as node 80, it should output like

**10->80: [19]**

if you input source node as ‘n’, it should output like

**-END-**

Example2: suppose you execute **spfw1**, you should see:

**Please input network filename:**(then you input, say, input\_100\_400\_1.sp)  
**Please input a source node** **[input 'n' to stop]:** (then you input, say, 2)

**Please input a sink node:** (then you input, say, 20)

then you should see the output like

**2->20: [9] 20<-19<-94<-1<-2**

if you input source node as node 10, sink node as node 80, it should output like

**10->80: [19] 80<-22<-21<-66<-10**

if you input source node as ‘n’, it should output like

**-END-**