```
      ∇CLOSE[⎕]∇
[0]   CLOSE
[1]   ⎕NUNTIE ¯1
      ∇COMMA[⎕]∇
[0]   Z←COMMA V
[1]   Z←⍕V
[2]   Z[(Z=' ')/⍳⍴Z]←','
      ∇CREATE[⎕]∇
[0]   CREATE F
[1]  ⍝ CREATE FILENAME AND TIE WITH 1
[2]   ('/Users/shallit/Desktop/',F)⎕NCREATE ¯1
      ∇DELDEAD[⎕]∇
[0]   Z←DELDEAD G;A;B;I;C;D
[1]  ⍝ G is a matrix in Grail format
[2]  ⍝ result is new Grail automaton with the single nonaccepting state
[3]  ⍝ deleted in all transitions
[4]   A←(∨/G='F')/[1]G
[5]   B←,A,' '
[6]   B←⍋(B∊'0123456789 ')/B
[7]   Z←(0,¯1↑⍴G)⍴''
[8]   I←0
[9]  L1:
[10]  I←I+1
[11]  →(I>1↑⍴G)/0
[12]  C←G[I;]
[13]  →(∨/C∊'SF')/L2
[14]  D←⍋C
[15]  →(~∧/D[1 3]∊B)/L1
[16] L2:
[17]  Z←Z,[1]C
[18]  →L1
```

```
      ∇GRAILTOGV[⎕]∇
[0]   Z←GRAILTOGV M;Z1;Z2;Z3;Z4;I;J;A;B;C;D;E;F;S;ST;N
[1]  ⍝ M is a matrix with each line a row of a grail spec of an automaton
[2]   Z1←'digraph G {',CR
[3]   Z1←Z1,'rankdir = LR;',CR
[4]   Z5←Z3←Z4←''
[5]   N←(⍴M)[1]
[6]   S←ST←⍳0
[7]   TR←(~∨/M∈'SF')/[1]M
[8]   TR2←⍕,TR,' '
[9]   TR3←(((⍴TR2)÷3),3)⍴TR2
[10]  TR4←∪⊂[2]TR3[;1 3]
[11]  I←0
[12] L0:
[13]  I←I+1
[14]  →(I>⍴TR4)/L7
[15]  A←⊃TR4[I]
[16]  B←(TR3[;1 3]∧.=A)/[1]TR3
[17]  Z5←Z5,(⍕A[1]),' -> ',(⍕A[2]),'[ label = "',(COMMA B[;2]),'"];',CR
[18]  →L0
[19] L7:
[20]  I←0
[21] L1:
[22]  I←I+1
[23]  →(I>N)/L4
[24]  A←RTB M[I;]
[25]  →((1⍬↑A)∧.='(START) |-')/L2
[26]  →((¯1⍬↑A)∧.='-| (FINAL)')/L3
[27]  →L1
[28] L3:
[29]  F←⍕¯1⍬↓A
[30]  S←S,F
[31]  Z3←Z3,'node [shape = doublecircle, label="',(⍕F),'"fontsize=12]',(⍕F),';',CR
[32]  →L1
[33] L2:
[34]  Z4←Z4,'node [shape = point ]; qi',CR
[35]  Z4←Z4,'qi ->0;',CR
[36]  →L1
[37] L4:
[38]  ST←∪ST
[39]  ST←(~ST∈S)/ST
[40]  J←0
[41] L5:
[42]  J←J+1
[43]  →(J>⍴ST)/L6
[44]  Z3←Z3,'node [shape = circle, label="',(⍕ST[J]),'", fontsize=12]',(⍕ST[J]),';',CR
[45]  →L5
[46] L6:
[47]  Z←Z1,Z3,Z4,Z5,'}',CR
```

```
      ∇GRAILTOMAPLE[⎕]∇
[0]   ZZ←GRAILTOMAPLE V;A;B;C;D;N;E;F;G;H;L;M;P;Q;R;S;T;U
[1]  ⍝ converts vector with grail representation to matrix rep
[2]   A←⎕AV[3]MATRIFY V
[3]   B←(~∧/A=' ')/[1]A
[4]   C←(~∨/B='(')/[1]B
[5]   D←⍉C,' '
[6]   N←(⍳0)⍴(⍴D)÷3
[7]   D←(N,3)⍴D
[8]   E←∪,D[;1 3]
[9]   ZZ←'m := Matrix(1..',(⍕⍴E),',1..',(⍕⍴E),',fill=0);',CR
[10]  F←(2⍴⍴E)⍴0
[11]  F SET2⊂[2]E⍳D[;1 3]
[12]  H←(∨/B='S')/[1]B
[13]  L←,H,' '
[14]  M←(L∊'0123456789 ')/L
[15]  P←⍉M
[16]  Q←(⍴E)⍴0
[17]  Q[E⍳P]←1
[18]  R←(∨/B='F')/[1]B
[19]  S←,R,' '
[20]  T←(S∊'0123456789 ')/S
[21]  U←⍉T
[22]  V←(⍴E)⍴0
[23]  V[E⍳U]←1
[24]  ZZ←ZZ,'u := Vector[row](',(⍕⍴E),',fill=0);',CR
[25]  ZZ←ZZ,'u[1] := 1;',CR
[26]  ZZ←ZZ,'v := Vector[column](',(⍕⍴E),',fill=0);',CR
[27]  I←0
[28] L1:
[29]  I←I+1
[30]  →(I>⍴U)/0
[31]  ZZ←ZZ,'v[',(⍕E⍳U[I]),'] := 1;',CR
[32]  →L1
      ∇ISPAL[⎕]∇
[0]   Z←ISPAL X
[1]   Z←∧/X=⌽X
```

```
      VMAKE[⎕]V
[0]   MAKE
[1]   2 MINNUMPAL 9
[2]   2 MINNUMPAL 10
[3]   2 MINNUMPAL 11
[4]   MAKEMAPLE'minpal2-11'
[5]   2 MINNUMPAL 13
[6]   3 MINNUMPAL 4
[7]   MAKEGV'minpal3-4'
[8]   3 MINNUMPAL 5
[9]   MAKEMAPLE'minpal3-5'
[10]  2 MINLENPAL 4
[11]  MAKEGVPDF'minlpal2-4'
[12]  2 MINLENPAL 5
[13]  MAKEGVPDF'minlpal2-5'
[14]  MAKEMAPLE'minlpal2-5'
[15]  3 MINLENPAL 1
[16]  MAKEGVPDF'minlpal3-1'
[17]  3 MINLENPAL 2
[18]  MAKEGVPDF'minlpal3-2'
[19]  MAKEMAPLE'minlpal3-2'
[20]  4 MINLENPAL 1
[21]  MAKEGVPDF'minlpal4-1'
[22]  MAKEMAPLE'minlpal4-1'
[23]  2 MINEOPAL 2 5
[24]  MAKEGVPDF'minpal2-e2o5'
[25]  MAKEMAPLE'minpal2-e2o5'
[26]  2 MINEOPAL 6 3
[27]  MAKEGVPDF'minpal2-e6o3'
[28]  MAKEMAPLE'minpal2-e6o3'
[29]  3 MINEOPAL 0 3
[30]  MAKEGVPDF'minpal3-e0o3'
[31]  MAKEMAPLE'minpal3-e0o3'
[32]  2 NUMEOPAL 3 9
[33]  2 NUMEOPAL 3 8
[34]  2 NUMEOPAL 4 7
[35]  2 NUMEOPAL 4 6
[36]  2 NUMEOPAL 5 5
[37]  2 NUMEOPAL 5 4
[38]  2 NUMEOPAL 6 5
[39]  2 NUMEOPAL 6 4
[40]  2 NUMEOPAL 7 4
[41]  2 NUMEOPAL 8 4
[42]  2 NUMEOPAL 3 10
[43]  2 NUMEOPAL 4 8
[44]  2 NUMEOPAL 5 6
[45]  2 NUMEOPAL 7 5
[46]  2 NUMEOPAL 9 4
[47]  3 NUMEOPAL 1 5
```

```
MINNUMPAL       NUMEOPAL        OPEN    PALLEN  PALSIN  QUEUE   QUEUE2  QUEUE3  QUEUE4  READ    RLB     RTB     SET2
      ∇MAKEGV[□]∇
[0]   MAKEGV F;M;C;D;E
[1]   ⍝ make a .gv and .pdf file from Grail file named in F
[2]   OPEN F,'.txt'
[3]   M←READ
[4]   CLOSE
[5]   C←⍕1 0↓□AV[3]MATRIFY M
[6]   D←DELDEAD C
[7]   E←GRAILTOGV D
[8]   CREATE F,'.gv'
[9]   WRITE E
[10]  CLOSE
[11]  C←□SH'/usr/local/bin/dot -Tpdf ~shallit/Desktop/',F,'.gv > ~shallit/Desktop/',F,'.pdf'
      ∇MAKEGVPDF[□]∇
[0]   MAKEGVPDF F;M;C;D;E
[1]   ⍝ make a .gv and .pdf file from Grail file named in F
[2]   OPEN F,'.txt'
[3]   M←READ
[4]   CLOSE
[5]   C←⍕1 0↓□AV[3]MATRIFY M
[6]   D←DELDEAD C
[7]   E←GRAILTOGV D
[8]   CREATE F,'.gv'
[9]   WRITE E
[10]  CLOSE
[11]  C←□SH'/usr/local/bin/dot -Tpdf ~shallit/Desktop/',F,'.gv > ~shallit/Desktop/',F,'.pdf'
      ∇MAKEMAPLE[□]∇
[0]   MAKEMAPLE F;M;C
[1]   OPEN F,'.txt'
[2]   M←READ
[3]   CLOSE
[4]   C←GRAILTOMAPLE M
[5]   CREATE F,'maple.txt'
[6]   WRITE C
[7]   CLOSE
      ∇MATRIFY[□]∇
[0]   Z←C MATRIFY X;I;D
[1]   I←(X=C)/⍳⍴X
[2]   D←(I,1+⍴X)-1+0,I
[3]   Z←((⍴D),⌈/D)⍴(,D∘.≥⍳⌈/D)\,(X≠C)/X
```

```
[0]   K MINEOPAL EO
[1]  A make automata for minimum number of palindromes
[2]   A←K QUEUE2 EO
[3]   CREATE D←'pal',(⍕K),'-e',(⍕EO[1]),'o',(⍕EO[2]),'.txt'
[4]   WRITE A
[5]   CLOSE
[6]   C←⎕SH grailpath,'fmmin ~shallit/Desktop/',D,' > ~shallit/Desktop/min',D
[7]   G←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/',D
[8]   E←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/min',D
[9]   'writing file ',D,'; ',(⍕G[3]),' states'
[10]  'writing file min',D,'; ',(⍕E[3]),' states'
[11]  ''
      ∇MINLENPAL[⎕]∇
[0]   K MINLENPAL N;A
[1]  A make automata for no pal of length N over alph of size K
[2]   A←K QUEUE3 N
[3]   CREATE D←'lpal',(⍕K),'-',(⍕N),'.txt'
[4]   WRITE A
[5]   CLOSE
[6]   C←⎕SH grailpath,'fmmin ~shallit/Desktop/',D,' > ~shallit/Desktop/min',D
[7]   G←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/',D
[8]   E←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/min',D
[9]   'writing file ',D,'; ',(⍕G[3]),' states'
[10]  'writing file minl',D,'; ',(⍕E[3]),' states'
[11]  ''
      ∇MINNUMPAL[⎕]∇
[0]   K MINNUMPAL N
[1]  A make automata for minimum number of palindromes
[2]   A←K QUEUE N
[3]   CREATE D←'pal',(⍕K),'-',(⍕N),'.txt'
[4]   WRITE A
[5]   CLOSE
[6]   C←⎕SH grailpath,'fmmin ~shallit/Desktop/',D,' > ~shallit/Desktop/min',D
[7]   G←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/',D
[8]   E←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/min',D
[9]   'writing file ',D,'; ',(⍕G[3]),' states'
[10]  'writing file min',D,'; ',(⍕E[3]),' states'
[11]  ''
      ∇NUMEOPAL[⎕]∇
[0]   K NUMEOPAL EO
[1]  A make automata for minimum number of palindromes
[2]   A←K QUEUE4 EO
[3]   CREATE D←'palnum',(⍕K),'-e',(⍕EO[1]),'o',(⍕EO[2]),'.txt'
[4]   WRITE A
[5]   CLOSE
[6]   C←⎕SH grailpath,'fmmin ~shallit/Desktop/',D,' > ~shallit/Desktop/min',D
[7]   G←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/',D
[8]   E←⍎⍕⎕SH grailpath,'fmstats ~shallit/Desktop/min',D
[9]   'writing file ',D,'; ',(⍕G[3]),' states'
[10]  'writing file min',D,'; ',(⍕E[3]),' states'
[11]  ''
```

```
      ∇OPEN[⎕]∇
[0]    OPEN F
[1]   ⍝ OPEN FILENAME F
[2]    ('/Users/shallit/Desktop/',F)⎕NTIE ¯1
      ∇PALLEN[⎕]∇
[0]    Z←PALLEN X;N;I;J;A
[1]   ⍝ lengths of nonempty palindromes occurring in string X
[2]    N←(⍴X)[1]
[3]    Z←N⍴0
[4]    I←¯1
[5]   L1:
[6]    I←I+1
[7]    →(I>N)/L3
[8]    J←0
[9]   L2:
[10]   J←J+1
[11]   →((I+J)>N)/L1
[12]   A←X[(I+1)TO I+J]
[13]   →(~ISPAL A)/L2
[14]   Z[J]←1
[15]   →L2
[16] L3:
[17]   Z←Z/⍳⍴Z
      ∇PALSIN[⎕]∇
[0]    Z←PALSIN V;N;L;I;T
[1]    Z←⍳0
[2]    N←(⍴V)[1]
[3]    L←0
[4]   L1:
[5]    L←L+1
[6]    →(L>N)/0
[7]    I←¯1
[8]   L2:
[9]    I←I+1
[10]   →((I+L)>N)/L1
[11]   T←V[I+⍳L]
[12]   →(T∨.≠⌽T)/L2
[13]   Z←Z,⊂T
[14]   →L2
```

```
      ∇QUEUE[⎕]∇
[0]   XX←K QUEUE N;A;Q;B;C;D;E;C2;F;J;L;I;⎕IO
[1]  ⍝ alphabet size K, at most N palindromes (incl empty word)
[2]  ⍝ the output is in Grail format
[3]   ⎕IO←1
[4]   N←N-1
[5]   XX←'(START) |- 1',CR
[6]   SEEN←Q←,⊂(⍳0)(⍳0)
[7]  L1:
[8]   →(0=⍴Q)/DONE
[9]   ⍝(0∈1000|(⍴Q),⍴SEEN)/'⎕←(⍴Q),⍴SEEN'
[10]  A←⊃Q[1]
[11]  Q←1↓Q
[12]  B←⊃A[1]
[13]  C←⊃A[2]
[14]  D←¯1
[15] L2:
[16]  D←D+1
[17]  →(D≥K)/L1
[18]  E←B,D
[19]  C2←SORT∪C,PALSIN E
[20]  E←(-(¯1+2×N)⌊⍴E)↑E
[21]  →((⍴C2)>N)/L2
[22]  F←E C2
[23]  →((⊂F)∈SEEN)/L3
[24]  Q←Q,⊂F
[25]  SEEN←SEEN,⊂F
[26] L3:
[27]  J←SEEN⍳⊂F
[28]  L←SEEN⍳⊂A
[29]  XX←XX,(⍕L),' ',(⍕D),' ',(⍕J),CR
[30]  →L2
[31] DONE:
[32]  I←0
[33] L4:
[34]  I←I+1
[35]  →(I>⍴SEEN)/0
[36]  XX←XX,(⍕I),' -| (FINAL)',CR
[37]  →L4
```

```
      ∇QUEUE2[⎕]∇
[0]   XX←K QUEUE2 EO;A;B;C;D;E;C2;CO;CE;F;Q;J;L;I
[1]   ⍝ alphabet size K, pals of max length EO[1] (even) and EO[2] (odd)
[2]   ⍝ states are last few symbols and list of pal lengths
[3]    XX←'(START) |- 1',CR
[4]    G←⌈/EO
[5]    SEEN←Q←,⊂(⍳0)(⍳0)
[6]   L1:
[7]    →(0=⍴Q)/DONE
[8]    A←⊃Q[1]
[9]    Q←1↓Q
[10]   B←⊃A[1]
[11]   C←⊃A[2]
[12]   D←¯1
[13]  L2:
[14]   D←D+1
[15]   →(D≥K)/L1
[16]   E←B,D
[17]   E←(-((G+2)⌊⍴E))↑E
[18]   C2←∪C,PALLEN E
[19]   CO←(1=2|C2)/C2
[20]   CE←(0=2|C2)/C2
[21]   →((∨/CE>EO[1])∨∨/CO>EO[2])/L2
[22]   F←E C2
[23]   →((⊂F)∈SEEN)/L3
[24]   Q←Q,⊂F
[25]   SEEN←SEEN,⊂F
[26]  L3:
[27]   J←SEEN⍳⊂F
[28]   L←SEEN⍳⊂A
[29]   XX←XX,(⍕L),' ',(⍕D),' ',(⍕J),CR
[30]   →L2
[31]  DONE:
[32]   I←0
[33]  L4:
[34]   I←I+1
[35]   →(I>⍴SEEN)/0
[36]   XX←XX,(⍕I),' -| (FINAL)',CR
[37]   →L4
```

```
      ∇QUEUE3[⎕]∇
[0]    XX←K QUEUE3 LEN;A;B;C;D;E;C2;CO;CE;F;Q;J;L;I
[1]   ⍝ alphabet size K, pals of max length LEN
[2]   ⍝ states are last few symbols and list of pal lengths
[3]    XX←'(START) |- 1',CR
[4]    G←LEN
[5]    SEEN←Q←,⊂(⍳0)(⍳0)
[6]   L1:
[7]    →(0=⍴Q)/DONE
[8]    A←⊃Q[1]
[9]    Q←1↓Q
[10]   B←⊃A[1]
[11]   C←⊃A[2]
[12]   D←¯1
[13]  L2:
[14]   D←D+1
[15]   →(D≥K)/L1
[16]   E←B,D
[17]   E←(-((G+2)⌊⍴E))↑E
[18]   C2←∪C,PALLEN E
[19]   →(∨/C2>LEN)/L2
[20]   F←E C2
[21]   →((⊂F)∈SEEN)/L3
[22]   Q←Q,⊂F
[23]   SEEN←SEEN,⊂F
[24]  L3:
[25]   J←SEEN⍳⊂F
[26]   L←SEEN⍳⊂A
[27]   XX←XX,(⍕L),' ',(⍕D),' ',(⍕J),CR
[28]   →L2
[29]  DONE:
[30]   I←0
[31]  L4:
[32]   I←I+1
[33]   →(I>⍴SEEN)/0
[34]   XX←XX,(⍕I),' -| (FINAL)',CR
[35]   →L4
```

```
      ∇QUEUE4[□]∇
[0]   XX←K QUEUE4 EO;A;Q;B;C;D;E;C2;F;J;L;I
[1]  ⍝ alphabet size K, at most EO[1] even pals and EO[2] odd pals (incl empty word)
[2]   N←+/EO
[3]   N←N-1
[4]   XX←'(START) |- 1',CR
[5]   SEEN←Q←,⊂(⍳0)(⍳0)
[6]  L1:
[7]   →(0=⍴Q)/DONE
[8]   ⍝(0∈1000|(⍴Q),⍴SEEN)/'□←(⍴Q),⍴SEEN'
[9]   A←⊃Q[1]
[10]  Q←1↓Q
[11]  B←⊃A[1]
[12]  C←⊃A[2]
[13]  D←¯1
[14] L2:
[15]  D←D+1
[16]  →(D≥K)/L1
[17]  E←B,D
[18]  C2←SORT∪C,PALSIN E
[19]  E←(-(¯1+2×N)⌊⍴E)↑E
[20]  DE←+/0=2|∈⍴¨C2
[21]  DO←+/1=2|∈⍴¨C2
[22]  →((DE>EO[1])∨DO>EO[2])/L2
[23]  F←E C2
[24]  →((⊂F)∈SEEN)/L3
[25]  Q←Q,⊂F
[26]  SEEN←SEEN,⊂F
[27] L3:
[28]  J←SEEN⍳⊂F
[29]  L←SEEN⍳⊂A
[30]  XX←XX,(⍕L),' ',(⍕D),' ',(⍕J),CR
[31]  →L2
[32] DONE:
[33]  I←0
[34] L4:
[35]  I←I+1
[36]  →(I>⍴SEEN)/0
[37]  XX←XX,(⍕I),' -| (FINAL)',CR
[38]  →L4
```

```
      ∇READ[⎕]∇
[0]   Z←READ
[1]   Z←⎕NREAD ¯1 82 ¯1
      ∇RLB[⎕]∇
[0]   Z←RLB V
[1]   Z←(¯1+(V=' ')⍳0)↓V
      ∇RTB[⎕]∇
[0]   Z←RTB V
[1]   Z←⌽RLB⌽V
      ∇SET2[⎕]∇
[0]   M SET2 L;I;A
[1]   I←0
[2]  L1:
[3]   I←I+1
[4]   →(I>1↑⍴L)/0
[5]   A←⊃L[I]
[6]   ZZ←ZZ,'m[',(⍕A[1]),',',(⍕A[2]),'] := ','m[',(⍕A[1]),',',(⍕A[2]),'] + 1;',CR
[7]   →L1
      ∇SORT[⎕]∇
[0]   Z←SORT V
[1]   Z←V[⍋V]
[2]   Z←Z[⍋∈⍴¨Z]
      ∇TO[⎕]∇
[0]   Z←A TO B
[1]   Z←(A-1)+⍳1+B-A
      ∇WRITE[⎕]∇
[0]   Z←WRITE X
[1]   Z←X ⎕NAPPEND ¯1
      
```