

```

1  !MS5-RC-Concat.f95
2  !2025.01.17.0955cst JMS- Concatenation & Order computations
3  ! Computer- "T4"/HP-800-G4-Mini/i7-8700T/IntelUHD630/win10Pro-22H2
4  !           ^name ^Mfgr.Id ^chipset ^graphics ^OS
5  !           /AbsoftProFortran 21.0.2/OpenGL+Glut3.6
6  !           ^compiler           ^Fortran graphics
7  !Table of Contents - ToC:
8  !Subroutine Concatenate(VtotL,VAnew,VAnext,VAPrev,iP)-opens Data-nLAV.txt
9  !                                           -opens Data-AttConcat.txt
10 !Subroutine AllMoves(VtotL,VAout,VAPrevin,nCin,iP) -Concatenation-based Emul.
11 !Subroutine ComputeOrder(VtotL,VAin,nOrder,iP) -order of Zone subsets
12 !-----7 9
13
14 Subroutine Concatenate(VtotL,VAnew,VAnext,VAPrev,iP)
15 !2025.01.17.0955cst JMS-
16 !           VAnew(1:VtotL) <- VAnext(1:VtotL) <- VAPrev(1:VtotL)
17 !The VA's (1:VtotL) must be in natural order.
18
19 !--Globals
20 use MS1Def ,only: Ur,Us,Ut
21 use MS2RCDef ,only: ASymbol
22 !--End Globals
23 implicit none
24 !--Arguments
25 integer(4)::VtotL
26 integer(4)::VAnew( 27) ! The votes [1:VtotL]
27 integer(4)::VAnext( 27)
28 integer(4)::VAPrev( 27)
29 integer(4)::iP !write enable>5: write(iP,...)
30 !--Internals
31 integer(4)::Init,nA1,nV,nL
32 integer(4)::AoutAAprev(1:24,1:24)
33 integer(4)::nLAV(1:24,0:27)
34 integer(4)::nVAPrev(27)
35 integer(4)::VAnextmod( 27)
36 integer(4)::VAnewL(20)
37 !--EndDefs-----
38
39 if(Init==0) then
40   if(iP>5) write(iP,"(/'Concatenate: Concat@L40'/)")
41   write(iP,"( 'Import Data-nLAV.txt:')")
42   !if(iP>5) write(iP,"('nLAV(1:24,0:27): nL = nLAV(A,V): transposed ')")
43   open(Ur, file='Data-nLAV.txt', action='read' &
44         , access='sequential' , status='old' )
45   read(Ur,"(//)") !Skips the first 3 lines
46   !if(iP>5) write(iP,"(24(2x,a1),' <-:nA')") (char(ASymbol(nV)),nV=1,24)
47   do nL=0,27
48     read(Ur,*) nLAV(1:24,nL)
49     !if(iP>5) write(iP,"(24i3)") nLAV(1:24,nL)
50   enddo!nL
51   !if(iP>5) write(iP,"(' ^:V')")
52   close(Ur)
53
54   !Import AoutAAprev(1:24,1:24) = "Attitude Concatenations"
55   if(iP>5) write(iP,"( 'Import Data-AttConcat.txt:')")
56   open(Ur, file='Data-AttConcat.txt', action='read' &
57         , access='sequential' , status='old' )
58   read(Ur,"(//)") !Skips the first 3 lines
59   do nA1=1,24
60     read(Ur,*) AoutAAprev(nA1,1:24)
61     !write(iP,"(24( i2,',')") AoutAAprev(nA1,1:24)
62     !!write(iP,"(24(1x,a1,',')") char(ASymbol(AoutAAprev(nA1,1:24)))

```

```

63     enddo!nA1
64     !if(iP>5) write(iP, &
65     !      "( ^:morph rotation changes to previous`s nL locations'")
66     close(Ur)
67
68     Init = 1
69     endif!(Init==0)
70
71     !Recurring processing: -----
72     VAnewL = 0
73     nVAprev = 0
74     if(iP==6) write(iP,*)
75
76     !Concatenation is accomplished in the next 7 lines of code:
77     do nV = 1,VtotL
78         nVAprev( nV) = nLAV(VAprev(nV),nV)
79         VANextmod( nV) = VANext(nVAprev(nV))
80         if(VAprev( nV) == 0) cycle
81         if(VANextmod(nV) == 0) cycle
82         VAnewL(nV) = AoutAAprev(VANextmod(nV),VAprev(nV))
83     enddo!nV
84
85     if(iP>5) then
86         write(iP,"(/'Concatenate in previous`s order: -----')")
87         write(iP,"( 'nV      :',:,20i3      )") ( nV      ,nV=1,VtotL)
88         write(iP,"(/'A VAprev  :',:,20i3,' :previous'") VAprev( 1:VtotL)
89         write(iP,"('      ""      :',:,20(2x,a1)")) (char(Asymbol(VAprev( 1:VtotL))))
90
91         write(iP,"(/'A VANext   :',:,20i3,' :pre-morph'") VANext( 1:VtotL)
92         write(iP,"('      ""      :',:,20(2x,a1)")) (char(Asymbol(VANext( 1:VtotL))))
93
94         write(iP,"(/'n nVAprev  :',:,20i3,' :morph'") nVAprev( 1:VtotL)
95         write(iP,"(/'A VANextmod: ',:,20i3,' :post-morphed'")VANextmod(1:VtotL)
96         write(iP,"('      ""      :',:,20(2x,a1)")) (char(Asymbol(VANextmod(1:VtotL))))
97
98         write(iP,"(/'A VAnew   :',:,20i3,' :concatenation'")VAnewL( 1:VtotL)
99         write(iP,"('      ""      :',:,20(2x,a1)")) (char(Asymbol(VAnewL( 1:VtotL))))
100    endif!(iP>5)
101
102    VAnew(1:VtotL) = VAnewL(1:VtotL)
103 End Subroutine Concatenate
104 !-----7 9
105
106 Subroutine AllMoves(VtotL,VAout,VAprevin,nCin,iP)
107 !2024.12.22.2130cst JMS-
108 !For nCin>0: Adds / Appends moves to the left end of VAprevin
109 !      VAout(1:VtotL) <- VA(nCin) <- VAprevin(1:VtotL)
110 !For nCin<0: Subtracts / Removes moves from the right end of VAprevin
111 !      VAout(1:VtotL) <- VAprevin(1:VtotL) <- VA(inverse of +nCin)
112 !The first implementation uses concatenation.
113
114 !--Globals
115 use MS1Def ,only: Ur,Us,Ut
116 use MS2RCDef ,only: Asymbol,Csymbol
117 !--End Globals
118 implicit none
119 !--Arguments
120 integer(4)::VtotL
121 integer(4)::VAout(20) !votes- out [1:VtotL]
122 integer(4)::VAprevin(20) ! - in
123 integer(4)::nCin !move choice- [-18,18]
124 integer(4)::iP !write enable>5: write(iP,...)
125 !--Internals

```

```

125 !--Internals
126 integer(4)::Init,nC
127 integer(4)::VAl1(20,-18:18)
128 integer(4)::nCinv(0:19)=(/0,2,1,4,3,6,5,8,7,10,9,12,11,13,14,15,16,17,18,19/)
129 integer(4)::VAprev(20)
130
131 !--EndDefs-----
132
133 if(Init==0) then
134   if(iP>5) write(iP,"(/'AllMoves():  Concat@L134'/)")
135   !Positive nC's add moves to the left/scramble side of VAprev:
136   VAl1(1:20, 0) = & !"aaaaaaaaaaaaaaaaaaaaa ", 0
137   (/ 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1/)
138   VAl1(1:20, 1) = & !"babababaabaababaabaa ", 1
139   (/ 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1/)
140   VAl1(1:20, 2) = & !"cacacacaacaacacaacaa ", 2
141   (/ 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1/)
142   VAl1(1:20, 3) = & !"ababababaabaababaaba ", 3
143   (/ 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2, 1, 2/)
144   VAl1(1:20, 4) = & !"acacacacaacaacacaaca ", 4
145   (/ 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3/)
146   VAl1(1:20, 5) = & !"ddaaddaadaaaddaadaaaa ", 5
147   (/ 4, 4, 1, 1, 4, 4, 1, 1, 4, 4, 1, 1, 4, 4, 1, 1, 4, 4, 1, 1/)
148   VAl1(1:20, 6) = & !"eaeaeaeaeaeaeaeaeae ", 6
149   (/ 5, 5, 1, 1, 5, 5, 1, 1, 5, 5, 1, 1, 5, 5, 1, 1, 5, 5, 1, 1/)
150   VAl1(1:20, 7) = & !"aaddaaddaadaaddaadaad ", 7
151   (/ 1, 1, 4, 4, 1, 1, 4, 4, 1, 1, 4, 4, 1, 1, 4, 4, 1, 1, 4, 4/)
152   VAl1(1:20, 8) = & !"aeaeaeaeaeaeaeaeaeae ", 8
153   (/ 1, 1, 5, 5, 1, 1, 5, 5, 1, 1, 5, 5, 1, 1, 5, 5, 1, 1, 5, 5/)
154   VAl1(1:20, 9) = & !"ffffaaafffffaaaaaaaa ", 9
155   (/ 6, 6, 6, 6, 1, 1, 1, 1, 6, 6, 6, 6, 1, 1, 1, 1, 1, 1, 1, 1/)
156   VAl1(1:20,10) = & !"ggggaaaagggggaaaaaaaa ",10
157   (/ 7, 7, 7, 7, 1, 1, 1, 1, 7, 7, 7, 7, 1, 1, 1, 1, 1, 1, 1, 1/)
158   VAl1(1:20,11) = & !"aaaafffffaaaaaaafffff ",11
159   (/ 1, 1, 1, 1, 6, 6, 6, 6, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 6, 6, 6, 6/)
160   VAl1(1:20,12) = & !"aaaagggggaaaaaaagggg ",12", -
161   (/ 1, 1, 1, 1, 7, 7, 7, 7, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 7, 7, 7, 7/)
162   VAl1(1:20,13) = & !"hahahahaahaahahaaha ",13
163   (/ 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1/)
164   VAl1(1:20,14) = & !"ahahahahaahaahahaha ",14
165   (/ 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8, 1, 8/)
166   VAl1(1:20,15) = & !"iaaiaaiaaiaaiaaiaa ",15
167   (/ 9, 9, 1, 1, 9, 9, 1, 1, 9, 9, 1, 1, 9, 9, 1, 1, 9, 9, 1, 1, 1, 1/)
168   VAl1(1:20,16) = & !"aaiiaaiiaaiaaiaai ",16
169   (/ 1, 1, 9, 9, 1, 1, 9, 9, 1, 1, 9, 9, 1, 1, 9, 9, 1, 1, 9, 9, 1, 1, 9/)
170   VAl1(1:20,17) = & !"jjjjaaaajjjjaaaaaaa ",17
171   (/10,10,10,10, 1, 1, 1, 1,10,10,10,10, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1/)
172   VAl1(1:20,18) = & !"aaaajjjjaaaaaaaajjjj ",18
173   (/ 1, 1, 1, 1,10,10,10,10, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,10,10,10,10/)
174   !Negative nC's undo moves from the right/solved side of VAprev:
175   do nC = 1,18
176     VAl1(1:20,-nC) = VAl1(1:20,nCinv(nC))
177   enddo
178   if(iP>5) then
179     do nC = -18,18
180       if(nC== -18) write(iP,"(/'Undo moves from the right/solved side:')")
181       if(nC< 0) write(iP,"(' ',a1,2x,20a1)") &
182         char(Csymbol(abs(nC))), char(Asymbol(VAl1(1:20,nC)))
183     enddo
184     if(nC== 0) write(iP,"(/'No change:  VAout=VAprevin')")
185     if(nC==0) write(iP,"(' ',a1,2x,20a1)") &
186       char(Csymbol(abs(nC))), char(Asymbol(VAl1(1:20,nC)))

```

```

187
188         if(nC== 1) write(iP,"(/'Add moves to the left/scrambled side:')")
189         if(nC> 0) write(iP,"('+',a1,2x,20a1)") &
190             char(Csymbol(abs(nC))), char(Asymbol(VAll(1:20,nC)))
191         enddo!nC
192     endif!(iP>5)
193     Init = 1
194 endif!(Init==0)
195
196 !Recurring processing: -----
197 VAprev = 0
198 VAprev(1:VtotL) = VAprevIn(1:VtotL)
199
200 select case(nCin)
201 case(:-19,19:) !nCin out of range
202     stop "Concat@L204 Allmoves: nCin out of range. Halt"
203
204 case( -18:-1) !Undo moves from the right/solved side:'
205     call Concatenate(VtotL,VAout,VAprev,VAll(1,nCin), 0)
206
207 case( 0 ) !No change: VAout=VAprevin
208     call Concatenate(VtotL,VAout,VAll(1,nCin),VAprev, 0)
209
210 case( 1:18) !Add moves to the left/scrambled side:
211     call Concatenate(VtotL,VAout,VAll(1,nCin),VAprev, 0)
212 end select!(nCin)
213 if(iP>5) write(iP,"( i2,2x,,:,20(2x,a1)") nCin,char(Asymbol(VAout(1:VtotL)))
214                                     return
215 End Subroutine AllMoves
216 !-----
217 !AllMoves(): Concat@223
218 !Undo moves from the right/solved side:
219 !-I aaaajjjjaaaaaaaaajjjj
220 !-H aaaajjjjaaaaaaaaajjjj
221 !-G aaiiaaiiaaiiaaiiaai
222 !-F iiaaiiaaiiaaiiaaiiaa
223 !-E ahahahahaahaahaaha
224 !-D hahahahaahaahaahaahaa
225 !-C aaaafffffaaaaaaafffff
226 !-B aaaagggggaaaaaaagggg
227 !-A ffffaaaafffffaaaaaaa
228 !-9 ggggaaaaggggaaaaaaa
229 !-8 aaddaaddaadaaddaadaad
230 !-7 aeeaeaeaeaeaeaeaeae
231 !-6 ddaaddaadaaddaadaadaa
232 !-5 eeaeaeaeaeaeaeaeae
233 !-4 ababababaabaababaaba
234 !-3 acacacacaacaacacaaca
235 !-2 babababaabaababaabaa
236 !-1 ababababaabaababaaba!
237 !No change: VAout=VAprevin
238 ! 0 aaaaaaaaaaaaaaaaaaaaa
239 !Add moves to the left/scrambled side:
240 !+1 babababaabaababaabaa
241 !+2 ababababaabaababaaba
242 !+3 ababababaabaababaaba
243 !+4 acacacacaacaacacaaca
244 !+5 ddaaddaadaaddaadaadaa
245 !+6 eeaeaeaeaeaeaeaeae
246 !+7 aaddaaddaadaaddaadaad
247 !+8 aeeaeaeaeaeaeaeaeae
248 !+9 ffffaaaafffffaaaaaaa
249 !+0 ggggaaaaggggaaaaaaa

```

```

249 !+A  yyyyyaaaayyyyaaaaaaaaa
250 !+B  aaaaafffffaaaaaaaafffff
251 !+C  aaaaggggaaaaaaaagggg
252 !+D  hahahahaahaahaahaa
253 !+E  ahahahahaahaahaaha
254 !+F  iiaaiaaiaaaiaaiaaa
255 !+G  aaiiaaiaaaiaaiaaai
256 !+H  jjjjaaaajjjjaaaaaaa
257 !+I  aaaajjjjaaaaaaaajjjj
258 !-----7 9
259
260 Subroutine ComputeOrder(VtotL,VAin,nOrder,iP)
261 !2025.01.17.1745cst JMS- Computes the order of scramble VAin(20)
262 !Testcase:
263 ! Ein%cVAtest      = "tdtfbdkadtffbdkadbaa"
264 ! Ein%VA(1: 8)     = 20, 4,20, 6, 2, 4,11, 1,
265 ! Ein%VA(9:20)    = 4,20, 6, 6, 2, 4,11, 1, 4, 2, 1, 1,
266 !                  "aiaqwlksaiaqjhisalwaa"  !:self-concat visually confirmed.
267
268 !--Globals
269 use MS1Def ,only: Ur,Us,Ut
270 use MS2RCDef ,only: ASymbol,Ztot,Z,EinW,cZrFilename,iDiscord !,nOrder
271 !--End Globals
272 implicit none
273 !--Arguments
274 integer(4)::VtotL
275
276 integer(4)::VAin(27)! The votes [1:VtotL]
277 integer(4)::nOrd(0:9)
278 integer(4)::iP !Write enable>5: write(iP,...)
279 !--Internals
280 integer(4)::VArroot(27)
281 integer(4)::nOrd(0:9),VAorder(27),VANext(27)
282 integer(4)::nV,iZ,nZ
283
284 !--EndDefs-----
285 if(iP>5) write(iP,"(/'ComputeOrder: Concat@L297'/)")
286 VArroot = 0
287 VArroot(1:VtotL) = VAin(1:VtotL)
288 VAorder = 0
289 VAorder = VArroot
290 nOrd = (/ 1,0,0,0,0,0,0,0,0,0 /)
291
292 10 continue
293 do iZ = 1,Ztot; nZ = Z(iZ)
294   if(nOrd(nZ)==0) then
295     do nV = 1,EinW(nZ)%Vtot
296       if(VAorder(EinW(nZ)%V(nV))/=1) goto 22
297     enddo!nV
298     nOrd(nZ) = nOrd(0)
299   endif!(nOrd(nZ)==0)
300 22 continue
301 enddo!nZ
302
303 !if(nOrd(0)>5) goto 30
304 do iZ = 1,Ztot; nZ = Z(iZ)
305   if(nOrd(nZ)==0) goto 40
306 enddo!nV
307 if(iP>5) write(iP,"(10i5,2x,.,20a1)") nOrd, char(ASymbol(VAorder(1:VtotL)))
308 call SaveOutFile
309 30 nOrder = nOrd ;return
310

```

```
311 40 if(iP>5) write(iP,"(10i5,2x,.,20a1)") nOrd, char(Asymbol(VAorder(1:VtotL)))
312   call Concatenate(VtotL,VAorder,VARoot,VAorder, 0)
313
314   nOrd(0) = nOrd(0) + 1
315   if(nOrd(0)>2000) stop'Computed order exceeded 2000. Stopping.'
316   goto 10
317 End Subroutine ComputeOrder
318 !-----7 9
319
320
```