```
1: ()
 2: Prgm
 3: DispG
 4: Local cx1,cy1,cx2,cy2,cxs,cys,asp,k,o,prevpic,redraw,t,tmpzoom,slots
 5: \{0,0\}\rightarrow k
 6: \{0,0\}\rightarrow 0
 7: StoPic prevpic
 8: (xmax-xmin)/158→cxs
 9: (ymax-ymin)/76→cys
10: 158/76→asp
11: xmin→cx1
12: ymax→cy1
13: xmax→cx2
14: ymin→cy2
15:
16: misc\statline("idle")
17:
18: If not isVar(ωprgmvrs\zooms) Then
19: newMat (10, 6) \rightarrow \omegaprgmvrs\zooms
20: EndIf
21: If string(\omega prgmvrs zooms[10]) = "[[0,0,0,0,0,0]]" Then
22: [[xmin,xmax,ymin,ymax,xscl,yscl]]→ωprgmvrs\zooms[10]
23: EndIf
24:
25: misc\statline("msg:Move the box to pan/zoom...")
26: true→redraw
27: Lbl drawbox
28: If redraw Then
29: RplcPic prevpic
30: Line cx1, cy1, cx2, cy1
31: Line cx2, cy1, cx2, cy2
32: Line cx2, cy2, cx1, cy2
33: Line cx1, cy2, cx1, cy1
34: false→redraw
35: EndIf
36: Goto finishbx
37:
38:
39: getKey() \rightarrow k[1]
40: While k[1] \neq 13 \ \text{@[ENTER]}
41: rotate(k)\rightarrowk
42: getKey() \rightarrow k[1]
43:
44: If k[1]=337 Then \mathcal{O}\leftarrow
45:
     If k[1]=k[2] Then
46:
       \{ -5, 0 \} \rightarrow 0
47:
      Else
48:
       \{ -1, 0 \} \rightarrow 0
49:
      EndIf
50:
51: ElseIf k[1]-4096=337 Then \mathcal{O}[2nd]+[\leftarrow]
52:
      \{-20,0\}\rightarrow 0
53:
54: ElseIf k[1]=340 Then
55:
     If k[1]=k[2] Then
56:
       \{5,0\}\to 0
57:
      Else
58:
        \{1,0\}\to 0
59:
      EndIf
60:
61: ElseIf k[1]-4096=340 Then \mathcal{O}[2nd]+[\rightarrow]
```

```
62:
       \{20,0\}\to 0
 63:
 64: ElseIf k[1]=344 Then \mathcal{O}_{\downarrow}
 65:
       If k[1]=k[2] Then
 66:
         \{0, -5\} \rightarrow 0
 67:
        Else
         \{0, 1\} \rightarrow 0
 68:
 69:
         EndIf
 70:
 71: ElseIf k[1] - 4096 = 344 Then O[2nd] + [\downarrow]
 72:
       \{0, -20\} \rightarrow 0
 73:
 74: ElseIf k[1]=338 Then \mathcal{O}_{\uparrow}
 75:
       If k[1]=k[2] Then
 76:
         \{0, 5\} \rightarrow 0
 77:
        Else
 78:
         \{0,1\}\to 0
 79:
        EndIf
 80:
 81: ElseIf k[1]-4096=338 Then O(2nd)+[f]
 82:
       \{0, 20\} \rightarrow 0
 83:
 84: ElseIf k[1]=339 Then \mathcal{O}_{\uparrow} \leftarrow
 85:
       If k[1]=k[2] Then
 86:
         \{ -5, 5\} \rightarrow 0
 87:
        Else
 88:
         \{ 1, 1 \} \rightarrow 0
 89:
         EndIf
 90:
 91: ElseIf k[1]=342 Then \mathcal{O}_{\uparrow}\rightarrow
 92:
       If k[1]=k[2] Then
 93:
         \{5, 5\} \to 0
 94:
       Else
 95:
         \{1, 1\} \to 0
 96:
         EndIf
 97:
 98: ElseIf k[1]=345 Then \mathcal{O}_{\downarrow} \leftarrow
 99:
       If k[1]=k[2] Then
100:
         \{ -5, -5 \} \rightarrow 0
101:
       Else
102:
         \{ -1, -1 \} \rightarrow 0
103:
        EndIf
104:
105: ElseIf k[1]=348 Then \mathcal{O}_{\downarrow} \rightarrow
106:
      If k[1]=k[2] Then
107:
         \{5, 5\} \to 0
108:
       Else
109:
         \{1, 1\} \to 0
110:
         EndIf
111:
112: ElseIf k[1]=43 Then O[+]
113:
      cx1-asp*cxs*5→cx1
114:
        cx2+asp*cxs*5→cx2
       cy1+cys*5→cy1
115:
116:
        cy2-cys*5→cy2
117:
        true→redraw
118:
119: ElseIf k[1]-4096=43 Then O(2nd)+[+]
120:
       cx1-asp*cxs→cx1
121:
       cx2+asp*cxs→cx2
122:
         cy1+cys→cy1
```

```
123:
       cy2-cys→cy2
124:
      true→redraw
125:
126: ElseIf k[1]=45 Then \mathcal{O}(-)
127:
     cx1+asp*cxs*5→cx1
128:
      cx2-asp*cxs*5→cx2
129:
     cy1-cys*5→cy1
      cy2+cys*5→cy2
130:
131:
       true→redraw
132:
133: ElseIf k[1]-4096=45 Then O[2nd]+[-]
134:
      cx1+asp*cxs→cx1
135:
      cx2-asp*cxs→cx2
136:
      cy1-cys→cy1
137:
      cy2+cys→cy2
      true→redraw
138:
139:
140: ElseIf k[1]=48 Then \bigcirc 0
141:
      [[xmin, xmax, ymin, ymax, xscl, yscl]]→tmpzoom
142:
      ωprgmvrs\zooms[10,1]→xmin
143:
      ωprgmvrs\zooms[10,2]→xmax
144:
      ωprgmvrs\zooms[10,3]→ymin
145:
      ωprgmvrs\zooms[10,4]→ymax
146:
      ωprgmvrs\zooms[10,5]→xscl
147:
      ωprgmvrs\zooms[10,6]→yscl
148:
     tmpzoom→ωprgmvrs\zooms[10]
149:
      true→redraw
150:
     Goto quit
151:
152: ElseIf k[1]=49 Then O1
153:
     xmin→cx1
154:
     ymax→cy1
155:
     xmax→cx2
156:
     ymin→cy2
157:
      true→redraw
158:
159: ElseIf k[1]=258 Then O[STO]
     misc\statline("busy")
160:
161:
     misc\statline("msg:Loading...")
       ""→slots
162:
163:
     For t, 1, 9
164:
       If string (\omegaprgmvrs\zooms[t])="[[0,0,0,0,0,0]]" Then
165:
        slots&","&"
                       Slot "&string(t)→slots
166:
       Else
        slots&","&" ♦ Slot "&string(t)→slots
167:
168:
       EndIf
169:
       EndFor
170:
171:
       misc\flib2("menu:Save current window"&slots)
172:
       misc\statline("idle")
173:
       If fl[1] \neq 0 Then
174:
       [[xmin,xmax,ymin,ymax,xscl,yscl]]→ωprgmvrs\zooms[fl[1]]
175:
       misc\statline("msg:√ Saved to slot "&string(fl[1]))
176:
       EndIf
177:
178:
     ElseIf k[1]=4354 Then
179:
      misc\statline("busy")
180:
       misc\statline("msg:Loading...")
181:
       ""→slots
182:
       For t, 1, 9
183:
        If string (\omega prgmvrs \setminus zooms[t]) = "[[0,0,0,0,0,0]]" Then
```

```
184:
         slots&","&"
                         Slot "&string(t)→slots
185:
       Else
186:
         slots&","&" ◆ Slot "&string(t)→slots
187:
        EndIf
188:
       EndFor
189:
190:
       misc\flib2("menu:Recall window"&slots)
191:
       misc\statline("idle")
192:
       If fl[1] \neq 0 Then
193:
        If wprgmvrs\zooms[f1[1],1]\gequerymvrs\zooms[f1[1],2] or wprgmvrs\zooms[f1[1],
     3]≥wprgmvrs\zooms[fl[1],4] Then
194:
         misc\statline("msg: ERROR: Invalid or blank slot.")
195:
        Else
196:
         [[xmin, xmax, ymin, ymax, xscl, yscl]]→ωprgmvrs\zooms[10]
197:
         wprgmvrs\zooms[fl[1],1]→xmin
198:
         \omegaprgmvrs\zooms[fl[1],2]\rightarrowxmax
199:
         wprgmvrs\zooms[fl[1],3]→ymin
200:
         \omegaprgmvrs\zooms[fl[1],4]\rightarrowymax
201:
         \omegaprgmvrs\zooms[fl[1],5]\rightarrowxscl
202:
         \omegaprgmvrs\zooms[fl[1],6]\rightarrowyscl
203:
         true→redraw
204:
         Goto quit
205:
       EndIf
206:
      EndIf
207:
208: ElseIf k[1]=257 Then @[BKSPC]
      misc\statline("busy")
209:
210:
       misc\statline("msg:Loading...")
211:
       ""→slots
212:
       For t, 1, 9
213:
       If string(\omega prgmvrs zooms[t]) = "[[0,0,0,0,0,0,0]]" Then
214:
         slots&","&"
                        Slot "&string(t)→slots
215:
        Else
         slots&","&" ◆ Slot "&string(t)→slots
216:
217:
       EndIf
218:
       EndFor
219:
       misc\statline("idle")
220:
       misc\flib2("menu:Slot to delete"&slots&",All Slots")
221:
       If fl[1] \neq 0 and fl[1] \neq 10 Then
222:
        [[0,0,0,0,0,0]] \rightarrow \omega prgmvrs \setminus zooms[fl[1]]
223:
       misc\statline("idle")
224:
       misc\statline("msg: ✓ Erased slot "&string(fl[1]))
225:
     EndIf
226:
      If fl[1]=10 Then
227:
       newMat(10,6)→ωprgmvrs\zooms
228:
        [[xmin,xmax,ymin,ymax,xscl,yscl]]→ωprgmvrs\zooms[10]
229:
       misc\statline("idle")
230:
       misc\statline("msg:√ Erased all zoom slots.")
231:
       EndIf
232:
      If fl[1]=0 Then
233:
       misc\statline("idle")
234:
       EndIf
235:
236: ElseIf k[1]=4353 Then O(INS)
237:
238:
        Title "Restore Backup Zooms"
239:
        Text "Restoring backup zooms will overwrite"
        Text "all current slots."
240:
        Text ""
241:
        Text "This cannot be undone."
242:
243:
       EndDlog
```

```
misc\statline("idle")
244:
245:
      If ok=1 Then
246:
       Try
247:
        wprgmvrs\zoomsbak→wprgmvrs\zooms
248:
        misc\statline("msg: ✓ Restored backup zooms.")
249:
      Else
250:
        ClrErr
251:
       misc\statline("msg: ERROR: Backup zooms not found.")
252:
      EndTry
253:
     EndIf
254:
255: ElseIf k[1]=264 or k[1]=263 Then @[ESC] or [CLEAR]
256:
     RplcPic prevpic
257:
      Goto quit
258:
259: ElseIf k[1]=277 Then O[HOME]
260:
     RplcPic prevpic
261:
     DispHome
262:
     Goto quit
263: EndIf
264:
265: If string(o) \neq "{0,0}" Then
266:
     cx1+o[1]*cxs→cx1
267:
     cx2+o[1]*cxs→cx2
268:
      cy1+o[2]*cys→cy1
269:
      cy2+o[2]*cys→cy2
270:
      \{0,0\}\to 0
271:
      true→redraw
272: EndIf
273:
274: Goto drawbox
275: Lbl finishbx
276: EndWhile:true→redraw
277:
278:
279: If xmin=cx1 and ymax=cy1 and xmax=cx2 and ymin=cy2 Then
280: false→redraw
281: RplcPic prevpic
282: Goto quit
283: EndIf
284:
285: If cx2<cx1 or cy1<cy2 Then
286: DelVar fl
287: RplcPic prevpic
288: misc\statline("msg: ERROR: Invalid zoom window. Try again")
289: Goto abort
290: Else
291: [[xmin,xmax,ymin,ymax,xscl,yscl]]-ωprgmvrs\zooms[10]
292: cx1→xmin
293: cx2→xmax
294: cy1→ymax
295: cy2→ymin
296: EndIf
297:
298: Lbl quit
299: DelVar fl
300: If redraw Then
301: misc\statline("busy")
302: ClrDraw
303: DispG
304: Else
```

```
305: misc\statline("clr")
306: EndIf
```

307:
308: Lbl abort
309: EndPrgm