



UNIR

FUNDAÇÃO UNIVERSIDADE FEDERAL DE RONDÔNIA - CAMPUS DE JI-PARANÁ
DEPARTAMENTO DE FÍSICA DE JI-PARANÁ – DEFUJI
X SEMANA DA FÍSICA 2017

Prof. Antonio F Cardozo



Programação Básica em Java



Minicurso 6

Java™



```
1
2 //MINICURSO JAVA BÁSICO 2017/2
3 //PROF ANTONIO F CARDOZO
4
5 import java.applet.Applet;
6 import java.applet.*;
7 import java.awt.*;
8 import java.lang.*;
9 import java.awt.geom.*;
10 import java.io.*;
11 import java.awt.event.*;
12 import java.awt.Graphics;
13 import java.util.Date;
14 import java.text.*;
15 import java.io.BufferedWriter;
16 import java.io.File;
17 import java.io.FileNotFoundException;
18 import java.io.FileWriter;
19 import java.util.Scanner;
20 import java.util.GregorianCalendar;
21 import java.text.SimpleDateFormat;
22 import javax.swing.JOptionPane;
23
24 public class quedal extends java.applet.Applet implements
25 Runnable, AdjustmentListener
26 {
27 Image image,cabe,PfundoQ;
28 double XPF=0;
29 double XPI=0;
30 String eventDesc = " ";
31 String teta=new String("\u03b8");
32
33 final static BasicStroke stroke0 = new BasicStroke(0.5f);
34 final static BasicStroke stroke1 = new BasicStroke(1.0f);
35 final static BasicStroke stroke2 = new BasicStroke(4.0f);
36 final static BasicStroke stroke3 = new BasicStroke(6.0f);
37
38 FontMetrics fontMetrics;
39 //DIALOG
40 Label DIaux1, DIaux2;
41 Panel DIp1,DIp2,DIp3;
42 TextArea DIta,DItb,DItc;
43 String af;
44 String data;
45
46 Frame window;
47 //DIALOG
48 Panel p2,p3, p4,p5;
49 Label Isnt, Isnt2, Isnt3,Isnt4 ,Isnt5;
50 TextField caixa1, caixa2, caixa3;
51 Button button1, button2;
52 Scrollbar sb1;
53 Choice ch,ch2;
54 Color azul= new Color(120, 170, 210);
55 Color verde=new Color(221, 245, 200);
56 Color cornova = new Color(010,010,060);
57
58 TextArea ta= new TextArea(5, 50);
59 NumberFormat nf1 = NumberFormat.getNumberInstance();
60 NumberFormat nf2 = NumberFormat.getNumberInstance();
61
62 Font f1, f2, f3;
63 String eventmessage;
```

```

64
65 public void init()
66 {
67     cabe = getImage(getDocumentBase(), "CabeQueda.jpg");
68     resize(1240,800); setLayout(new BorderLayout()); setBackground(Color.gray);
69     f1= new Font("Arial", Font.PLAIN, 14);
70     f2= new Font("Bodoni MT", Font.ITALIC, 10);
71
72     Calcular= new Button("Calcular");Calcular.setFont(f2);
73     Limpar= new Button("Limpar");
74
75     aux1= new Label("V0 (m/s)");aux1.setBackground(Color.darkGray);
76     aux1.setForeground(corTextoClaro);aux1.setFont(f1);
77     aux2= new Label("R=b*v");aux2.setBackground(Color.darkGray);
78     aux2.setForeground(corTextoClaro);aux2.setFont(f1);
79
80     caixa1= new TextField("0",2); caixa1.setForeground(Color.black);
81     caixa2= new TextField("0",1); caixa2.setForeground(Color.black);
82     ch = new Choice ( );ch.setFont(f1);ch.setBackground(Color.darkGray);
83     ch.setForeground(corTextoClaro);
84     ch2 = new Choice ( );ch2.setFont(f1);ch2.setBackground(Color.darkGray);
85     ch2.setForeground(corTextoClaro);
86
87     ch2.add("Ocultar Dados");
88     ch2.add("Mostrar Variáveis");
89     ch2.add("Ocultar Todas");
90
91
92     add("East", p4);
93     p4.setBackground(verde);
94     p4.setLayout(new GridLayout(2,1));
95     add("South", p2);
96     p2.setBackground(Color.darkGray);
97     p2.setLayout(new GridLayout(2,10));
98     sb1 = new Scrollbar(Scrollbar.HORIZONTAL,1,0,0,7);
99     sb1.addAdjustmentListener(this);
100    sb1.setBackground(Color.darkGray);sb1.setForeground(corTextoClaro);
101    caixa4.setFont(f1);
102    p2.add(aux1);//m
103    p2.add(aux2);//k
104    p2.add(aux3);//r
105    p2.add(caixa1); caixa1.setText("60");caixa1.setFont(f1);//v0
106    p2.add(caixa2); caixa2.setText("0");caixa2.setFont(f1);//r0
107    p2.add(caixa3); caixa3.setText("0");caixa3.setFont(f1);//x0
108
109    p4.add(ta);ta.setBackground(Color.black);
110    ta.setForeground(corVerdeClaro); ta.setEditable(false); ta.setFont(f3);
111    p4.add(p3);////////////////////
112    p3.setLayout(new GridLayout(20,1));
113    p3.add(aux10);aux10.setFont(f1);
114    p3.add(Calcular); Calcular.setFont(f2);
115    Apagar1();
116 } //init
117
118 public boolean mouseUp (Event evt, int x, int y)
119 {
120     POSIX=x;POSIY=y;
121     if (evt.modifiers == Event.META_MASK)
122     { infop=3;
123     eventmessage = "P(" + (x-40) + ", " + (500-y) + ")"; repaint();
124     }
125     //Apagar1();grafico1();
126     return true;

```

```
127 }
128 public boolean mouseMove(Event e, int x, int y)
129 {
130     infop=1;
131     POSIX=x;POSIY=y;
132     return true;
133 }
134
135 public boolean mouseDrag(Event evt, int x, int y)
136 {
137     infop=2;
138     nf1.setMaximumFractionDigits(2);
139     nf5.setMaximumFractionDigits(0);
140     stop();
141
142     POSIX=(double)(x);
143     POSIY=(double)(y);
144
145     POSIX0=POSIX;POSIY0=POSIY;
146     return false;
147 }
148
149 public boolean action (Event evt, Object alqum)
150 {
151     if (evt.target == Calcular)
152     {
153         t=0;ss1=1;ss2=1;ss3=1;ss4=1;
154         inicio=inicio+1;
155         stop();
156         infop=-2;
157         String s2 = getch2(); Dimension dm = size();
158         if (s2.compareTo("Mostrar Variáveis") == 0)
159         {
160             ocultar=0;
161             caixa1.setVisible (true);
162             caixa2.setVisible (true);
163         }
164
165         A0 = Double.valueOf(caixa5.getText()).doubleValue();
166         r=R0*2; thp = 0.01745329252*A0; i=0; m =1;
167         TN=0;
168         ta.setText("");
169         start();
170         return true;
171     }
172     else if (evt.target == avaliar)
173     {
174         window.show();
175         return true;
176     }
177     else if (evt.target == Parar)
178     {
179         stop();//TN=3;
180         return true;
181     }
182     else if (evt.target == Cont)
183     {
184         tex= Double.valueOf(caixa4.getText()).doubleValue();
185         start();
186         return true;
187     }
188     else if (evt.target == Limpar)
189     {
```

```

190     Apagar1();
191     //JOptionPane.showMessageDialog(null, "Texto Gravado");
192     return true;
193 }
194 else return false;
195 }
196
197 public void adjustmentValueChanged(AdjustmentEvent e)
198
199 {
200     if(e.getAdjustable() == sb1)
201     {
202         pxf=((double) sb1.getValue());
203         ESCS = pxf;
204         grafico7();
205     }
206 }
207
208 public String getch(){return ch.getSelectedItem();}
209 public String getch2(){return ch2.getSelectedItem();}
210
211 public void run()
212 {
213     while (Thread.currentThread() == clockThread)
214     {
215         nf6.setMaximumFractionDigits(4);
216         TN=TN+0.2;
217         if(TN>3000){ta.setText("");TN=0;}
218         x=x+vx*dt; y=y+vy*dt;
219         ax = -r*vx/ m;
220         ay = (-m*GR - r*vy)/m;
221         vx = vx + ax * dt; vy=vy+ay*dt;
222         try {clockThread.sleep(1);}
223         catch (InterruptedException e){}
224     }
225 }
226 FontMetrics pickFont(Graphics2D g2, String longString,int xSpace)
227 {
228     boolean fontFits = false;
229     Font font = g2.getFont();
230     FontMetrics fontMetrics = g2.getFontMetrics();
231     int size = font.getSize();
232     String name = font.getName();
233     int style = font.getStyle();
234     return fontMetrics;
235 }
236
237 public void Apagar1()
238 {
239
240     Graphics g = getGraphics();
241     g.drawImage(PfundoQ, 0, 0, this);
242
243 }
244
245 public void grafico1()
246 {
247     Graphics g = getGraphics();
248     g.setColor(Color.red);
249     g.drawString(teta + " = "+ nf5.format(ponteiro) , 720, 420);
250     if (ESCS!=5)
251     {
252         g.drawImage(image, 678, 76, this);

```

```

253 g.drawImage(PfundoQ, 0, 0, this);
254
255 //mostrador
256 for(ptp = ponteiro - 5D; ptp < ponteiro + 5D; ptp += 0.5D)
257 {
258     RaioEx = 140;
259     angle = ptp;
260
261     RaioEx = 140; angle = ptp; AngleRad = angle* 3.1415 / 180;
262     g.drawLine((int) (posix0), (int) (posiy0), (int) (posix1), (int) (posiy1));
263 }
264 if((y+Y0)>500-MAUSY3 && ss3<0 )
265 {
266     if(sinal>0){ss3=1;}
267     g.drawString("t="+ nf1.format(t), (int) (MAUSX), (int) (MAUSY-80));
268 }//s3
269
270 if(ESCS<3)
271 {
272     g.setColor(Color.green);
273     g.drawLine((int)MAUSX, (int) (MAUSY-40), (int) (MAUSX), (int)MAUSY+80);//s1
274     g.fillOval((int) (MAUSX+10), (int) (MAUSY-30), 20, 20);
275     g.setColor(Color.yellow);
276     g.drawLine((int)MAUSX2, (int)MAUSY2-40, (int) (MAUSX2), (int)MAUSY2+80);//s2
277     g.fillOval((int) (MAUSX2+10), (int) (MAUSY2-30), 20, 20);
278
279     g.setColor (Color.red);
280     g.drawString("S1", (int) (MAUSX+13), (int) (MAUSY-16));
281     g.drawString("S2", (int) (MAUSX2+13), (int) (MAUSY2-16));
282     g.setColor (Color.lightGray);
283     g.drawLine(50,500,1000,500);
284     g.drawLine(40,0,40,500);//y
285 }
286 }//if
287 }
288
289 public void grafico3()
290 {
291     Graphics g = getGraphics();
292     g.setColor (Color.red);
293     g.drawString(teta + " = "+ nf5.format(ponteiro) , 720, 420);
294     if (ESCS!=5){g.drawImage(image, 678, 76, this);}
295     //mostrador
296     g.drawImage(PfundoQ, 0, 0, this);
297
298     for(ptp = ponteiro - 5D; ptp < ponteiro + 5D; ptp += 0.5D)
299     {
300         RaioEx = 140;
301         angle = ptp;
302
303         RaioEx = 140; angle = ptp; AngleRad = angle* 3.1415 / 180;
304         posix0 = (posix + (RaioEx - 10) * Math.cos(AngleRad));
305         posiy0 = (posiy - (RaioEx - 10) * Math.sin(AngleRad));
306         posix1 = (posix + (RaioEx - 50) * Math.cos(AngleRad));
307         posiy1 = (posiy - (RaioEx - 50) * Math.sin(AngleRad));
308         g.drawLine((int) (posix0), (int) (posiy0), (int) (posix1), (int) (posiy1));
309     }
310
311     if(ESCS==3 || ESCS==4)
312     {
313         g.setColor (Color.darkGray);
314         g.drawLine((int)MAUSX3-40, (int)MAUSY3, (int) (MAUSX3+40), (int)MAUSY3);//s3
315         g.fillOval((int) (MAUSX3+10), (int) (MAUSY3-30), 20, 20);

```

```

316 g.setColor(Color.orange);
317 g.drawLine((int)MAUSX4-40, (int)MAUSY4, (int)(MAUSX4+40), (int)MAUSY4); //s4
318 g.fillOval((int)(MAUSX4+10), (int)(MAUSY4-30), 20, 20);
319 g.setColor(Color.red);
320 g.drawString("S3", (int)(MAUSX3+13), (int)(MAUSY3-16));
321 g.drawString("S4", (int)(MAUSX4+13), (int)(MAUSY4-16));
322
323 g.setColor(Color.lightGray);
324 //g.setStroke(stroke2);
325 g.drawLine(50, 500, 1000, 500);
326 g.drawLine(40, 0, 40, 500); //y
327 }
328 }
329 public void grafico4()
330 {
331 Graphics g = getGraphics();
332 g.setColor(Color.darkGray);
333 g.fillOval((int)(POSIX), (int)(POSIY), 4, 4);
334 }
335
336 public void grafico5()
337 {
338 Graphics g = getGraphics();
339
340 g.setFont(f3);
341 g.setColor(Color.green);
342 g.clearRect((int)(POSIX), (int)(POSIY), 10, 10);
343 }
344 ///////////////////////////////////////////////////
345 public void grafico6()
346 {
347 Graphics g = getGraphics();
348 g.setColor(Color.red);
349 if (ESCS!=5){g.drawImage(image, 678, 76, this);}
350
351 for(ptp = ponteiro - 5D; ptp < ponteiro + 5D; ptp += 0.5D)
352 {
353 RaioEx = 140;
354 angle = ptp;
355
356 RaioEx = 140; angle = ptp; AngleRad = angle* 3.1415 / 180;
357 g.drawLine((int)(posix0), (int)(posiy0), (int)(posix1), (int)(posiy1));
358 }
359 g.drawString(teta + " = "+ nf5.format(ponteiro) , 830, 315);
360 }
361 public void grafico7()
362 {
363 Graphics g = getGraphics();
364 g.setFont(f1);
365 g.clearRect(570, 620, 100, 100);
366
367 }
368
369 public void paint(Graphics g)
370 {
371
372 if(sinal<0){stop();TN=0;}
373 Graphics2D g2 = (Graphics2D) g;
374 Dimension d = getSize();
375 fontMetrics = pickFont(g2, "Filled and Stroked GeneralPath", 8);
376 Color fg3D = Color.black; //Color.lightGray;
377 g.setColor(corTextoClaro);
378

```

```

379 PRX=440;
380 PRY=-20;
381 g.drawImage(image, 678, 76, this);
382
383 nf1.setMaximumFractionDigits(3);
384 nf2.setMaximumFractionDigits(3);
385 posix0=910;
386 posiy0=500;
387 g2.setStroke(stroke1);
388 g.drawImage(cabe,80, 0, this);
389 g.setColor(Color.black);
390
391 if(sinal>0)
392 {
393     //calculo
394     g.setFont(f1);
395     if(y+Y0<0)
396     {
397         g.drawString("t= "+nf1.format(t) + "s", (10), 60));
398         stop();sinal=-2;
399     }
400 }
401 if(t>tex)
402 {
403     g.drawString("t= "+nf1.format(t) + "s", 54, 30);
404     stop();sinal=-2;
405 }
406 g.fillOval((int)(X0+40+x), (int)(501-(y+Y0)),1,1);////graficay(t)
407 g.setColor(Color.green);
408 g.drawOval((int)(X0+40+t), (int)(501-vy),1,1);//////////graficav(t)
409 if(sinal<0){ta.append(nf2.format(x) +"\n");}
410 ///////////////posições
411 if(ESCS ==0)
412 {
413     mmv1 = Math.abs(X0 + x);
414     mmv2 = Math.abs(PAX);
415
416     mmv1 = Math.abs(y + Y0);
417     mmv2 = Math.abs(PAY);
418
419     mmv1 = Math.abs(vy);
420     mmv2 = Math.abs(PAV);
421
422 }
423 }//sinal
424
425 g2.setStroke(stroke2);
426 g.setColor(Color.white);
427 g.setFont(f1);
428 g.drawString("y(m)", (int)(PRX+340), (int)(PRY+400));
429 g.setColor(Color.red);
430
431
432 }
433
434 public void update (Graphics g) { paint (g);}
435 Thread clockThread = null;
436 public void start()
437 {
438 if (clockThread == null)
439 {
440 lockThread = new Thread(this, "Clock");clockThread.start();}
441 }

```



```
442 public void stop()  
443 {  
444 if(clockThread != null) {clockThread.stop();clockThread = null;}  
445 }  
446}  
447
```