

## Códigos criados no ChatGPT

### AutoLISP para uma cadeira:

```
(defun c:chair ()
  (setq height 40.0) ; height of the chair
  (setq width 30.0) ; width of the seat
  (setq depth 30.0) ; depth of the seat
  (setq leg_height 30.0) ; height of the legs
  (setq leg_width 5.0) ; width of the legs
  (setq backrest_height 20.0) ; height of the backrest
  (setq backrest_width 20.0) ; width of the backrest

  ; create seat
  (command "rectang" "0,0" (strcat width "," depth))

  ; create backrest
  (command "rectang" (strcat "0," (rtos depth)) (strcat backrest_width "," (rtos (+ depth
backrest_height))))

  ; create legs
  (command "rectang" "0,0" (strcat leg_width "," (rtos leg_height)))
  (command "rectang" (strcat (rtos width "," "0") "," (rtos leg_height)) (strcat (rtos (+ width
leg_width)) "," (rtos (+ depth leg_height))))
  (command "rectang" (strcat "0," (rtos (+ depth leg_height))) (strcat leg_width "," (rtos (+ depth
leg_height leg_height))))
  (command "rectang" (strcat (rtos width "," (rtos (+ depth leg_height))) "," (rtos (+ depth
leg_height))) (strcat (rtos (+ width leg_width)) "," (strcat (rtos (+ depth leg_height leg_height))
"," (rtos (+ depth leg_height leg_height)))))

  (princ)
)
```

### AutoLISP para uma mesa quadrada:

```
(defun c:table ()
  (setq table_width 30.0) ; width of the table
  (setq table_height 30.0) ; height of the table

  ; create table top
  (command "rectang" "0,0" (strcat (rtos table_width) "," (rtos table_height)))

  ; create legs
  (setq leg_width 5.0) ; width of the legs
  (setq leg_height 25.0) ; height of the legs
  (command "rectang" "0,0" (strcat (rtos leg_width) "," (rtos leg_height)))
  (command "rectang" (strcat (rtos (+ table_width leg_width)) "," "0") (strcat (rtos (+ table_width
leg_width leg_width)) "," (rtos leg_height)))
  (command "rectang" (strcat "0," (rtos (+ table_height leg_height))) (strcat (rtos leg_width) "," (rtos
(+ table_height leg_height leg_height)))
  (command "rectang" (strcat (rtos (+ table_width leg_width)) "," (rtos (+ table_height leg_height)))
(strcat (rtos (+ table_width leg_width leg_width)) "," (strcat (rtos (+ table_height leg_height
leg_height)) "," (rtos (+ table_height leg_height)))))

  (princ)
)
```

### AutoLISP para uma janela de 60x120cm:

```
(defun c:window ()
  (setq window_width 60.0) ; width of the window
  (setq window_height 120.0) ; height of the window
  (setq frame_thickness 5.0) ; thickness of the frame

  ; create window frame
  (command "rectang" "0,0" (strcat (rtos window_width) "," (rtos window_height)))

  ; create frame segments
  (command "rectang" (strcat (rtos frame_thickness) "," (rtos frame_thickness)) (strcat (rtos (-
window_width frame_thickness)) "," (rtos frame_thickness)))
  (command "rectang" (strcat (rtos (- window_width frame_thickness)) "," (rtos frame_thickness))
(strcat (rtos (- window_width frame_thickness)) "," (rtos (- window_height frame_thickness))))
  (command "rectang" (strcat (rtos frame_thickness) "," (rtos (- window_height frame_thickness)))
(strcat (rtos (- window_width frame_thickness)) "," (rtos (- window_height frame_thickness))))
  (command "rectang" "0,0" (strcat (rtos frame_thickness) "," (rtos window_height)))

  (princ)
)
```

### AutoLISP para uma porta de 90x210cm:

```
(defun c:door ()
  (setq door_width 90.0) ; width of the door
  (setq door_height 210.0) ; height of the door
  (setq frame_thickness 5.0) ; thickness of the frame

  ; create door panel
  (command "rectang" "0,0" (strcat (rtos door_width) "," (rtos door_height)))

  ; create door frame
  (setq frame_width (1.5 * frame_thickness)) ; width of the frame segments
  (command "rectang" (strcat (rtos (- frame_thickness)) "," (strcat (rtos (- frame_thickness)) "," (rtos
door_height))) (strcat (rtos (+ door_width frame_thickness)) "," (strcat (rtos (- frame_thickness))
"," (rtos (- frame_height frame_thickness)))))
  (command "rectang" "0,0" (strcat (rtos frame_thickness) "," (strcat (rtos door_height) "," (rtos
frame_thickness)))))
  (command "rectang" (strcat (rtos (- door_width frame_thickness)) "," (strcat (rtos (-
frame_thickness)) "," (rtos door_height))) (strcat (rtos (+ door_width frame_thickness)) "," (strcat
(rtos door_height) "," (rtos (- frame_thickness)))))
  (command "rectang" (strcat (rtos frame_thickness) "," (strcat (rtos (- door_height
frame_thickness)) "," (rtos door_width))) (strcat (rtos (- frame_thickness)) "," (strcat (rtos (-
frame_thickness)) "," (rtos (- door_height frame_thickness)))))
  (command "rectang" (strcat (rtos (+ door_width frame_thickness)) "," (strcat (rtos (- door_height
frame_thickness)) "," (rtos door_width))) (strcat (rtos door_width) "," (strcat (rtos (-
frame_thickness)) "," (rtos (- door_height frame_thickness)))))

  (princ)
)
```

AutoLISP para uma “porta” (garagem, módulo) de 290x500x240 cm:

```
(defun c:door ()
  (setq door_width 290.0) ; width of the door
  (setq door_height 240.0) ; height of the door
  (setq frame_thickness 500.0) ; thickness of the frame

  ; create door panel
  (command "rectang" "0,0" (strcat (rtos door_width) ", " (rtos door_height)))

  ; create door frame
  (setq frame_width (1.5 * frame_thickness)) ; width of the frame segments
  (command "rectang" (strcat (rtos (- frame_thickness)) ", " (strcat (rtos (- frame_thickness)) ", " (rtos
door_height))) (strcat (rtos (+ door_width frame_thickness)) ", " (strcat (rtos (- frame_thickness))
", " (rtos (- door_height frame_thickness)))))
  (command "rectang" "0,0" (strcat (rtos frame_thickness) ", " (strcat (rtos door_height) ", " (rtos
frame_thickness)))))
  (command "rectang" (strcat (rtos (- door_width frame_thickness)) ", " (strcat (rtos (-
frame_thickness)) ", " (rtos door_height))) (strcat (rtos (+ door_width frame_thickness)) ", " (strcat
(rtos door_height) ", " (rtos (- frame_thickness)))))
  (command "rectang" (strcat (rtos frame_thickness) ", " (strcat (rtos (- door_height
frame_thickness)) ", " (rtos door_width))) (strcat (rtos (- frame_thickness)) ", " (strcat (rtos (-
frame_thickness)) ", " (rtos (- door_height frame_thickness)))))
  (command "rectang" (strcat (rtos (+ door_width frame_thickness)) ", " (strcat (rtos (- door_height
frame_thickness)) ", " (rtos door_width))) (strcat (rtos door_width) ", " (strcat (rtos (-
frame_thickness)) ", " (rtos (- door_height frame_thickness)))))

  (princ)
)
```