NetLogo — Guia Completo de Primitivos (PT-PT)
Versão alvo: NetLogo 7.0.0 (Setembro 2025)

Este guia reúne praticamente todos os primitivos (comandos e repórteres) do core do NetLogo, organizados por categorias de utilização, com descrições e exemplos. Inclui ainda um índice alfabético e notas de versão.

Fontes de referência: Manual oficial do NetLogo (Dictionary) e documentação 7.0.0.

Gerado em: 27-09-2025

Como ler este guia

• *Primitivo*: palavra-chave embutida do NetLogo. Há **comandos** (não retornam valor) e **repórteres** (retornam valor).
• *Agentes*: `observer`, `turtles`, `patches` e `links`. Muitos primitivos só fazem sentido num tipo de agente.
• *Conjunto de agentes (agentset)*: conjunto não ordenado de agentes (ex.: `turtles`, `patches with [pcolor > 0]`).
• Notação dos exemplos:
  - Comentários começam com `;;`.
  - Exemplos assumem mundo padrão (topologia toroidal), salvo indicação.
• Este guia visa **NetLogo 7.0.0**. Algumas diferenças podem existir face a 6.x (ver “Notas de versão”).
• Extensões (`extensions [...]`) **não** estão exaustivamente aqui; focamo-nos no core.

Dica: Para saber “quem” pode usar um primitivo (observer/turtle/patch/link), consulta a entrada oficial no *Dictionary*.

1) Estrutura e controlo de fluxo

• to / to-report / end — comando
  Define procedimentos e repórteres.
  Exemplo:
  ```
  to setup
    clear-all
  end

  to-report media [xs]
    report mean xs
  end
  ```

- if / ifelse — comando
  Executa condicionalmente um bloco.
  Exemplo:
  ```
  if any? turtles [ ask turtles [ rt random 360 fd 1 ] ]
  ifelse count turtles > 100 [ user-message "muitas tartarugas!" ] [ crt 10 ]
  ```

- repeat / while / loop — comando
  Ciclos de repetição; `loop` requer `stop` dentro.
  Exemplo:
  ```
  repeat 10 [ fd 1 ]
  while [count turtles < 100] [ crt 1 ]
  loop [ if ticks > 100 [ stop ] tick ]
  ```

- foreach / map / filter / reduce — comando/repórter
  Itera/lista sobre listas.
  Exemplo:
  ```
  foreach [1 2 3] [ x -> show (x * x) ]
  set xs map [ x -> x * x ] [1 2 3]
  show filter [ x -> x > 2 ] xs
  show reduce + xs
  ```

- carefully [...] [handler] — comando
  Tenta executar; se der erro, corre handler sem interromper o modelo.
  Exemplo:
  ```
  carefully [ file-open "dados.txt" ] [ user-message "ficheiro em falta" ]
  ```

- stop / report — comando
  Termina procedimento; em repórteres usa-se `report`.
  Exemplo:
  ```
  to go
    if not any? turtles [ stop ]
    tick
  end
  ```

- run / runresult — comando/repórter
  Executa código passado como string/lista; use com cuidado.
  Exemplo:
  ```
  run "crt 1"
  show runresult "count turtles"
  ```

- without-interruption — comando
  Garante execução sem yield entre agentes (evita alternância).
  Exemplo:

```
  ask turtles [ without-interruption [ fd 1 rt random 60 ] ]
```

2) Contexto de agentes e conjuntos (agentsets)

• ask — comando
  Faz agentes executar comandos. Só agentes existentes à partida são chamados.
  Exemplo:
  ask turtles with [energy < 5] [set color red]

• of — repórter
  Puxa valores de agentes / agentsets.
  Exemplo:
  show [energy] of turtles
  show mean [pxcor] of patches

• with — repórter
  Filtra agentsets por predicado.
  Exemplo:
  set hungry-turtles turtles with [energy < 3]

• one-of / n-of — repórter
  Seleciona 1 ou N agentes aleatórios dum agentset.
  Exemplo:
  ask one-of turtles [die]
  set sample n-of 10 turtles

• other — repórter
  Todos os agentes excepto `self` dentro do mesmo conjunto.
  Exemplo:
  ask turtles [set near other turtles in-radius 1]

• min-one-of / max-one-of — repórter
  Escolhe o agente com menor/maior valor de um repórter.
  Exemplo:
  ask min-one-of turtles [energy] [set color blue]

• with-min / with-max — repórter
  Subconjunto com mínimo/máximo para um repórter.
  Exemplo:
  ask turtles with-max [energy] [set label "TOP"]

• in-radius / in-cone — repórter
  Agentes num raio/cone em torno de `self`,

Exemplo:
ask turtles [set neigh turtles in-radius 2]

• link-neighbors / link-neighbor? — repórter
Vizinhança definida por ligações.
Exemplo:
ask turtles [if any? link-neighbors [set color green]]

• myself / self / myself-of — repórter
Refere o agente actual (ou externo em `ask`).
Exemplo:
ask turtles [ask patch-here [set pcolor [color] of myself]]

3) Criação e remoção de agentes

• crt / create-turtles N [...] — comando
Cria tartarugas (opcionalmente com bloco de inicialização).
Exemplo:
```
crt 100 [
  setxy random-xcor random-ycor
  set color one-of base-colors
]
```

• hatch N [...] — comando
Clona `self` N vezes (herda variáveis).
Exemplo:
ask turtles with [energy > 10] [hatch 1 [set energy energy / 2]]

• sprout N [...] (patches) — comando
Cria tartarugas a partir de patches.
Exemplo:
ask patches with [pcolor = green] [sprout 2 [set color green]]

• die — comando
Remove o agente actual.
Exemplo:
ask turtles with [energy <= 0] [die]

• create-link-with / create-links-to / create-links-from — comando
Cria ligações (não dirigidas ou dirigidas conforme o ‘breed’).
Exemplo:
ask turtles [if random-float 1 < 0.1 [create-link-with one-of other turtles]]

- create-ordered-link-with — comando
  Cria link ordenado (use em breeds dirigidos).
  Exemplo:
  ask one-of turtles [create-ordered-link-with one-of other turtles]

4) Movimento e orientação (turtles)

- fd / bk — comando
  Avança/recua em unidades de patch.
  Exemplo:
  fd 1 ;; avança um patch
  bk 0.5 ;; recua meio patch

- rt / lt — comando
  Roda à direita/esquerda em graus.
  Exemplo:
  rt random 360

- setxy / move-to — comando
  Posiciona coordenadas ou move para agente/patch.
  Exemplo:
  setxy 0 0
  move-to patch 5 -2

- jump — comando
  Move em linha recta mantendo heading.
  Exemplo:
  jump 3

- face / facexy / towards / towardsxy — comando/repórter
  Aponta para agente/coords; `towards*` devolve heading em graus.
  Exemplo:
  facexy 0 0
  set heading towardsxy mouse-xcor mouse-ycor

- patch-ahead / patch-left-and-ahead / patch-right-and-ahead — repórter
  Patch em frente/esquerda/direita a uma distância.
  Exemplo:
  if [pcolor] of patch-ahead 1 = black [rt 90]

- can-move? / distance / distancexy — repórter
  Verifica movimento e mede distância (toroidal por defeito).
  Exemplo:

```
  if can-move? 1 [ fd 1 ]
  if distancexy 0 0 > 10 [ set color red ]
```

• dx / dy — repórter
  Componentes de deslocação dado `heading`.
  Exemplo:
  set xcor xcor + dx set ycor ycor + dy

5) Patches (grade)

• patch / patches — repórter
  Devolve patch em (x,y) ou o agentset de todos os patches.
  Exemplo:
  ask patch 0 0 [set pcolor green]

• patch-here / patch-at / neighbors / neighbors4 — repórter
  Acesso local aos patches vizinhos (8 ou 4).
  Exemplo:
  ask turtles [if any? neighbors with [pcolor = red] [rt 180]]

• pxcor / pycor / min-pxcor / max-pxcor / world-width / world-height — repórter
  Coordenadas e dimensão do mundo.
  Exemplo:
  show (list min-pxcor max-pxcor world-width)

• pcolor / plabel / plabel-color — variáveis
  Cor e rótulo do patch.
  Exemplo:
  ask patches [set plabel pxcor]

• sprout N [...] — comando
  Cria tartarugas neste patch (ver secção 3).

6) Links (redes)

• links / my-links / my-in-links / my-out-links — repórter
  Conjuntos de ligações associados a um turtle.
  Exemplo:
  ask turtles [show count my-links]

• tie / untie — comando
  Atar/Desatar: mover o turtle move o linkado.

Exemplo:
ask turtle 0 [tie]

• link-neighbors / link-neighbor? / distance-to — repórter
Vizinhos via rede; `distance-to` mede nº de passos na rede.
Exemplo:
ask turtles [set degree count link-neighbors]

• end1 / end2 / link-length / thickness / label / color — variáveis
Propriedades de links.
Exemplo:
ask links [set thickness 0.3 set color gray]

7) Variáveis e escopo

• globals [...] — declaração
Declara variáveis globais do observador.
Exemplo:
globals [temperature steps]

• turtles-own / patches-own / links-own [...] — declaração
Variáveis específicas por tipo de agente.
Exemplo:
turtles-own [energy age]

• breed [wolves wolf] / directed-link-breed [...] / undirected-link-breed [...] — declaração
Define raças (turtles) e tipos de links.
Exemplo:
breed [sheep a-sheep]
directed-link-breed [follows follow]

• set / let — comando
Atribuições; `let` cria variáveis locais.
Exemplo:
let choice one-of ["A" "B"]
set energy energy - 1

• reset-ticks / tick / tick-advance — comando
Ciclo temporal discreto/contínuo.
Exemplo:
reset-ticks
tick
tick-advance 0.25

8) Entrada/saída e mensagens ao utilizador

• print / show / type — comando
  Saída para o Monitor/Output.
  Exemplo:
  show count turtles
  type "Tartarugas: " print count turtles

• user-message / user-input / user-yes-or-no? / user-one-of — comando/repórter
  Caixas de diálogo de interação.
  Exemplo:
  if user-yes-or-no? "Criar mais 10?" [crt 10]

• file-open / file-close / file-print / file-write / file-read / file-read-line / file-type — E/S ficheiros
  Leitura e escrita em ficheiros textuais.
  Exemplo:
  file-open "out.csv"
  file-print (word "ticks," ticks)
  file-close

• file-exists? / file-at-end? / file-delete / file-flush — repórter/comando
  Estado e gestão de ficheiros.

• export-world / import-world — comando
  Exporta/Importa estado completo do modelo.

• export-plot / export-all-plots / export-view / export-interface / import-drawing / import-pcolors — comando
  Entrada/saída de gráficos e vista.

9) Gráficos (plots)

• set-current-plot / set-current-plot-pen — comando
  Selecciona plot/pen activo.
  Exemplo:
  set-current-plot "Pop"
  set-current-plot-pen "sheep"

• create-temporary-plot-pen — comando
  Cria pen temporária.

Exemplo:
create-temporary-plot-pen "run"

• plot / plotxy / histogram — comando
Plota valores; histogram gera distribuição.
Exemplo:
plot count sheep
plotxy ticks count wolves
histogram [energy] of turtles

• clear-plot / clear-all-plots / set-plot-x-range / set-plot-y-range / auto-plot? / set-plot-pen-mode / plot-pen-up / plot-pen-down — comando/repórter
Gestão de gráficos.

10) Matemática e comparações

• + - * / ^ mod — repórter
Operadores aritméticos.

• abs sqrt exp ln log10 — repórter
Funções matemáticas comuns (em graus para trig).

• sin cos tan asin acos atan — repórter
Trigonometria em graus.

• round floor ceiling int precision — repórter
Arredondamentos e precisão.

• min max mean median variance standard-deviation — repórter
Estatística básica.

• random random-float random-normal random-exponential random-poisson — repórter
Aleatoriedade (determinada por seed do modelo).

• = != < > <= >= — repórter
Comparadores.

• and or not — repórter
Lógica booleana.

11) Listas e dados estruturados

• list / sentence / word — repórter
  Construtores de listas/strings.
  Exemplo:
  set xs list 1 2 3
  set ys sentence xs [4 5] ;; concatena

• length first last but-first but-last item sublist — repórter
  Acesso/partes de listas.

• lput fput remove remove-item replace-item member? position — repórter
  Manipulação de listas.

• n-values range n-of shuffle sort sort-by — repórter
  Geração/ordenação/aleatoriedade.

• map foreach filter reduce — repórter/comando
  Transformações funcionais.

• table:... (extensão) — nota
  Para dicionários, usar extensão `table`.

12) Strings e conversões

• word substring length upper-case lower-case — repórter
  Operações de strings.

• read-from-string write-to-string — repórter
  Parse/serialização simples.

• is-number? is-string? is-list? is-agent? is-agentset? is-turtle? is-patch? is-link? is-boolean? — repórter
  Testes de tipo.

13) Cores

• color / pcolor / lcolor — variável
  Cor de turtles/patches/links.

• scale-color color value min max — repórter
  Mapeia valores para cores.
  Exemplo:
  set color scale-color green energy 0 10

- base-colors approximate-hsb extract-rgb extract-hsb — repórter
  Ferramentas de cor.

14) Interface, desenho e visão

- clear-all / clear-turtles / clear-patches / clear-drawing / clear-output — comando
  Limpeza do mundo, desenho e output.

- no-display / display — comando
  Suspende/retoma renderização para acelerar.

- pen-down (pd) / pen-up (pu) / pen-erase / pen-size / pen-mode — comando/variável
  Desenho com a caneta da tartaruga.
  Exemplo:
  ask turtles [pd set pen-size 0.3 fd 1 pu]

- inspect / watch / follow / reset-perspective — comando
  Ferramentas de inspeção/seguimento.

15) Mundo e topologia

- reset-ticks / tick / tick-advance — comando
  Tempo do modelo (ver secção 7).

- random-xcor / random-ycor — repórter
  Coordenadas aleatórias válidas.

- wrap-color? / patch-size / ticks-per-second — repórter
  Preferências/ambiente.

- set-patch-size / resize-world / set-world-... (via interface) — nota
  Configuração do mundo é normalmente feita na Interface, não via primitivos.

16) Depuração e diagnóstico

- print / show / type — comando
  Saída textual (ver secção 8).

- trace / untrace (6.x) / reset-timer / timer — comando/repórter
  Temporização grossa; `trace` foi descontinuado em versões recentes.

  Exemplo:
  reset-timer
  ; ... corre algo ...
  show timer

• error — comando
  Lança erro explícito (para validar pré-condições).
  Exemplo:
  if count turtles = 0 [error "Sem tartarugas"]

17) HubNet (multiutilizador) — resumo

• hubnet-reset / hubnet-broadcast / hubnet-send / hubnet-message-waiting? / hubnet-message — comandos/repórteres
  Suporte para atividades HubNet (fora do âmbito deste guia).

18) Notas de versão (7.0.0) — diferenças chave face a 6.4

• Novidades relevantes — nota
  NetLogo 7 introduziu alterações de interface, melhorias de documentação e paridade com NetLogo Web em alguns primitivos de diálogo (ex.: `user-input` com valor por omissão). Ver notas oficiais.

ÍNDICE ALFABÉTICO (seleção abrangente)
---
abs, acos, agentset, all?, and, any?, approx-hsb, asin, ask, atan, at-points,
back (bk), base-colors, behavior-space-run-number, breed, but-first, but-last,
can-move?, ceiling, clear-all, clear-drawing, clear-output, clear-patches,
clear-plot, clear-turtles, color, cos, count, create-link-with, create-links-from,
create-links-to, create-ordered-link-with, create-temporary-plot-pen, create-turtles (crt),
die, distance, distancexy, display, dx, dy, end1, end2, error, exp, extract-rgb, extract-hsb,
face, facexy, fd, file-at-end?, file-close, file-delete, file-exists?, file-flush, file-open,
file-print, file-read, file-read-line, file-type, file-write, filter, first, floor, follow,
fput, foreach, hatch, heading, histogram, if, ifelse, import-drawing, import-pcolors,
import-world, in-cone, in-radius, inspect, int, is-agent?, is-agentset?, is-boolean?, is-list?,
is-number?, is-patch?, is-string?, is-turtle?, item, jump, label, last, layout-circle, layout-spring,
left (lt), length, let, link-length, link-neighbor?, link-neighbors, links, ln, log10, lput,
map, max, max-one-of, mean, median, member?, min, min-one-of, mode-plot-pen (set-plot-pen-mode),

mod, move-to, my-in-links, my-links, my-out-links, myself, n-of, n-values, neighbors, neighbors4,
no-display, not, of, one-of, other, patch, patch-ahead, patch-at, patch-here, patch-left-and-ahead,
patch-right-and-ahead, pcolor, pen-down (pd), pen-up (pu), pen-erase, pen-size, pen-mode, plabel,
plabel-color, plot, plotxy, plot-pen-down, plot-pen-up, position, precision, print, pxcor, pycor,
random, random-float, random-normal, random-exponential, random-poisson, random-xcor, random-ycor,
range, reduce, remove, remove-duplicates, remove-item, repeat, report, replace-item, reset-perspective,
reset-ticks, reset-timer, right (rt), round, run, runresult, scale-color, self, sentence, set, setxy,
set-current-plot, set-current-plot-pen, set-plot-pen-mode, set-plot-x-range, set-plot-y-range,
shade-of?, shape, show, shuffle, sin, sort, sort-by, sprout, sqrt, standard-deviation, stop,
sublist, substring, table:* (extensão), tan, tick, tick-advance, ticks, timer, to, to-report, towards,
towardsxy, true, false, turtles, turtles-here, turtles-on, turtles-own, type, undirected-link-breed,
directed-link-breed, untie, upper-case, lower-case, user-message, user-input, user-yes-or-no?,
user-one-of, variance, watch, with, with-max, with-min, word, world-width, world-height, write-to-string,
read-from-string.