

1. Jakie dwie standardowe komendy są częścią prawie każdego modelu NetLogo?

- a) `setup, ask`
- b) `setup, go`
- c) `fd, rt`
- d) `set color, go`

2. Jaka instrukcja informuje interpreter na końcu każdej pętli w procedurze `go` o zakończeniu aktualnego kroku czasowego?

- a) `reset-ticks`
- b) `ca`
- c) `next-step`
- d) `tick`

3. Która z poniższych procedur każe wykonać coś wszystkim żółwiom na aktualnym polu?

- a) `ask turtles with [turtles-here] [do-something]`
- b) `ask turtles [do-something]`
- c) `ask turtles-patch [do-something]`
- d) `ask turtles-here [do-something]`

4. Dany jest program:

```
set mylist [7 5 3 2 1]
show (n-of 3 mylist)
```

Który rezultat działania programu jest możliwy?

- a) `[9 6 3]`
- b) `[7 2 1]`
- c) `[3]`
- d) `[6]`
- e) `[9]`

5. Wyjaśnij działanie:

```
show count [turtles-at 1 1] of patch 1 2
```

6. Wyjaśnij działanie:

```
ask turtles [
  if not any? turtles-on neighbors [
    set isolated? true
  ]
]
```

7. Wyjaśnij, do czego może służyć następująca procedura:

```
ask patches with [count neighbors != 8] [...]
```

8. Wyjaśnij działanie, jeśli świat ma rozmiar 11x11:

```
show max-n-of 5 patches with [pycor = 0] [pxcor]
```

9. Wyjaśnij działanie:

```
if shade-of? ( [pcolor] of patch-ahead 1 ) yellow [ ... ]
```

10. Która z poniższych procedur nie ma sensu interakcji?

- a) `ask n-of 100 turtles [set color color + who]`
- b) `repeat 1000 [set global-counter global-counter + 2]`
- c) `create-turtles 500 [setxy random 12 random 12]`
- d) `let name "agatka"`
- e) `breed[cows cow]`

11. Niech świat ma rozmiar 32 x 32. Która z poniższych procedur utworzy więcej niż 20 żółwi?

- a) `create-turtles 20 [forward 13]`
- b) `create-turtles 2 [forward 5 hatch 10 [forward 5]]`
- c) `ask n-of 20 patches [sprout 1 [set color red]]`
- d) `ask one-of turtles [hatch 20 [setxy random-pxcor random-pycor]]`
;zakładając, że jest już jeden żółw
- e) `repeat 20 [ask patches [sprout 1]]`

12. Jaś chce pokolorować na zielono 4 pola zajęte przez żółwie. Która procedura mu to umożliwi?

- a) `if 4 turtles-here [set pcolor green]`
- b) `ask n-of 4 patches with [any? turtles-here] [set pcolor green]`
- c) `ask 4 patches with [turtles-on] [set color green]`
- d) `repeat 4 [ask patches with [any? turtles-here] [set color green]]`
- e) `ask 4 patches [set pcolor green if not any? Turtles-here set pcolor black]`

13. Małgosia utworzyła suwak, który umożliwia ustawienie wartości zmiennej n w zakresie od 1 do 100. Chce utworzyć n żółwi losowo rozmieszczonych na ekranie, ale tak, by na jednym polu nie było więcej niż jednego żółwia. Która procedura jej to umożliwi?

- a) `create-turtles n [setxy random-pxcor random-pycor]`
- b) `create-turtles n [forward random]`
- c) `ask n-of n patches [sprout 1]`
- d) `create-turtles n [move-to one-of patches with [not any? turtles-here]`
- e) `create-turtles n turtles [move randomly until no turtles-here]`

14. W poniższym kodzie, jaka wartość n jest wyświetlona w okienku komend?

```
let n 7
while (n > 1)
[ set n (n - 2)
]
show n
```

- a) -2
- b) 1
- c) 2
- d) 5
- e) 7

15. W funkcji rekurencyjnej rekurencja kończy się, gdy pewien warunek jest spełniony. W poniższym programie, która linia za to odpowiada?

```
to spiral-recursive [stepStop]
  if (stepStop < 0.01)
    [ die
    ]
  forward stepStop
  right 45
  spiral-recursive (stepStop * 0.9)
end
```

- a) to spiral-recursive [stepStop]
- b) if (stepStop < 0.01)
- c) forward stepStop
- d) spiral-recursive (stepStop * 0.9)
- e) end

16. Jaki jest rezultat działania poniższej procedury?

```
ask turtles with [color = red]
[
  set pcolor brown
  hatch-nagrobki 1
  die
]
```

- a) Wszystkie czerwone żółwie zginą.
- b) Czerwone żółwie pozostawią po sobie ślad w postaci rasy nagrobków
- c) Po śmierci czerwonych żółwi pole, na którym stały, zmieni kolor na brązowy.
- d) Wszystkie żółwie zginą.
- e) a, b oraz c

17. Jaki jest wynik działania programu:

```
let %ct 0 repeat 50 [show %ct set %ct (%ct + 1) stop]
```

18. Dla twardzieli. Jaki jest sens następującej funkcji:

```
to-report tajemnicza-funkcja [x]
  report reduce * n-values x [i + 1]
end
```

19. Znalazłeś/aś kawałek papieru z częścią kodu:

```
Breed [humans human]
Breed [non-humans non-human]
Humans-own [ alive? Have-ammo? Agentset-of-friends]
Non-humans-own [ #-eaten am-I-intact? Need-brains?]
```

Napisz, czego model próbował napisać programista?