20)*alder(alder(s)) = s (2009ko apirila #1) -- #*

a) ++:: ([t], [t]) → [t]

[] ++ s = s (#1)
(x:r) ++ s = x:(r ++ s) (#2)

b) alder:: ([t]) → [t]

alder([]) = [] (#3)
alder(x:r) = alder(r) ++ (x:[]) (#4)

c) s-ren gainean indukzioa aplikatuz honako propietate hau frogatuko da orain:

alder(alder(s)) = s

Oinarrizko kasua edo kasu sinplea: s = []

alder(alder([])) = []?

✓ alder(alder([])) =(#3)
= alder([]) =(#3)
= []

Betetzen da. Berdintzaren alde biek balio bera dute.

Kasu orokorra edo induktiboa: s = x:r
alder(alder(x:r)) = x:r?

Indukzio-hipotesia: (r zerrendak propietatea bete egiten du)alder(alder(r)) = r

Frogatu nahi dugunera itzuliz: alder(alder(x:r)) = x:r?

✓ alder(alder(x:r)) =(#4)
= alder(alder(r) ++ (x:[])) =(Prop)
= alder(x:[]) ++ alder(alder(r)) =(#4)
= (alder([]) ++ (x:[]))++ alder(alder(r)) =(#3)
= ([] ++ (x:[]))++ alder(alder(r)) =(#1)
= (x:[]) ++ alder(alder(r)) =(#2)
= x:([] ++ alder(alder(r))) =(#1)
= x:(alder(alder(r))) =(hi)
= x:r

Beraz propietatea bete egiten da.

Urrats bakoitzean lerroaren eskuineko ertzean zehazten den ekuazioa formularen zein zatiri aplikatu zaion garbiago ikusteko koloreak erabili dira. Irakurgarritasunagatik bi kolore desberdin erabili dira baina kolore desberdinek ez dute inolako esanahi berezirik.

21)*luzera(hond(s)) = luzera(azkena_kendu(s)) (2009ko apirila #2) -- #*

a) luzera :: ([t]) → Int

luzera([]) = 0 (#1)
luzera(x:r) = 1 + luzera(r) (#2)

b) azkena_kendu :: ([t]) → [t]

azkena_kendu([]) = error "Zerrenda hutsa" (#3)
azkena_kendu(x:r)
| hutsa_da(r) = [] (#4)
| otherwise = x: azkena_kendu(r) (#5)

c) hond:: ([t]) → [t]

hond([]) = error " Zerrenda hutsa" (#6)
hond(x:r) = r (#7)

d) s-ren gainean indukzioa aplikatuz honako propietate hau frogatuko da orain:

luzera(hond(s)) = luzera(azkena_kendu(s))?

Oinarrizko kasua edo kasu sinplea: s = z:[]

luzera(hond(z:[])) = luzera(azkena_kendu(z:[]))?

✓ luzera(hond(z:[])) =(#7)
= luzera([]) =(#1)
= 0

✓ luzera(azkena_kendu(z:[])) =(#4)
= luzera([]) =(#1)
= 0

Betetzen da. Berdintzaren alde biek balio bera dute.

Kasu orokorra edo induktiboa: s = x:r (r zerrenda hutsa ez dela jakinda)

luzera(hond(z:r)) = luzera(azkena_kendu(z:r))?

Indukzio-hipotesia: (r zerrendak propietatea bete egiten du)

luzera(hond(r)) = luzera(azkena_kendu(r))?

Frogatu nahi dugunera itzuliz:

luzera(hond(z:r)) = luzera(azkena_kendu(z:r))?

✓ luzera(hond(z:r)) =(#6)
= luzera(r) =(Prop)
= 1 + luzera(hond(r)) =(hi)
= 1 + luzera(azkena_kendu(r))

✓ luzera(azkena_kendu(z:r)) =(#5)
= luzera(z:azkena_kendu(r)) =(#2)
= 1 + luzera(azkena_kendu(r))

Beraz propietatea bete egiten da.

Urrats bakoitzean lerroaren eskuineko ertzean zehazten den ekuazioa formularen zein zatiri aplikatu zaion garbiago ikusteko koloreak erabili dira. Irakurgarritasunagatik bi kolore desberdin erabili dira baina kolore desberdinek ez dute inolako esanahi berezirik.

22)aldiz(x, s) ≥ aldiz(x, azkena_kendu(s)) (2009ko ekaina) -- #

a) aldiz:: (t, [t]) → Int

```
aldiz(x, []) = 0                                         (#1)
aldiz(x, y:s)
        |x == y        = 1 + aldiz(x, s)                 (#2)
        |otherwise     = aldiz(x, s)                     (#3)
```

b) azkena_kendu:: ([t]) → [t]

```
azkena_kendu([]) = error "Zerrenda hutsa"                (#4)
azkena_kendu(x:r)
        | hutsa_da(r)        = []                        (#5)
        | otherwise          = x: azkena_kendu(r)        (#6)
```

c) Orain s zerrendaren gainean indukzioa aplikatuz eta s hutsa ez dela jakinda, honako propietate hau frogatuko da:

aldiz(x, s) ≥ aldiz(x, azkena_kendu(s))

Oinarrizko kasua: s = z:[]

aldiz(x, z:[]) ≥ aldiz(x, azkena_kendu(z:[]))?

Bi aukera edo azpikasu daude: x = z izatea edo x ≠ z izatea

- **x = z**

  ✓ aldiz(x, z:[]) =$^{(\#2)}$
  = 1 + aldiz(x, []) =$^{(\#1)}$
  = 1 + 0 = 1

  ✓ aldiz(x, azkena_kendu(z:[])) =$^{(\#5)}$
  = aldiz(x, []) =$^{(\#1)}$
  = 0

  1 ≥ 0 denez propietatea bete egiten da azpikasu honetan.

- **x ≠ z**

  ✓ aldiz(x, z:[]) =$^{(\#3)}$
  = aldiz(x, []) =$^{(\#1)}$
  = 0

  ✓ aldiz(x, azkena_kendu(z:[])) =$^{(\#5)}$
  = aldiz(x, []) =$^{(\#1)}$
  = 0

  0 ≥ 0 denez azpikasu honetan ere propietatea bete egiten da.

Beraz oinarrizko kasuan propietatea bete egiten da.

Kasu orokorra: s = z:r, r zerrenda hutsa ez dela jakinda

aldiz(x, z:r) ≥ aldiz(x, azkena_kendu(z:r))?

Indukzio-hipotesia: (x elementuak eta r zerrendak propietatea bete egiten dute)

aldiz(x, r) ≥ aldiz(x, azkena_kendu(r))

Frogatu nahi denera itzuliz:

aldiz(x, z:r) ≥ aldiz(x, azkena_kendu(z:r))?

Bi azpikasu edo aukera daude: x = z izatea edo x ≠ z izatea

- **x = z**

  - ✓ aldiz(x, z:r) $=^{(\#2)}$
    = 1 + aldiz(x, r)

  - ✓ aldiz(x, azkena_kendu(z:r)) $=^{(\#6)}$
    = aldiz(x, z:azkena_kendu(r)) $=^{(\#2)}$
    = 1 + aldiz(x, azkena_kendu(r))

    Indukzio-hipotesiagatik honako hau bete egiten da

    aldiz(x, r) ≥ aldiz(x, azkena_kendu(r))

    eta ondorioz honako beste hau ere bete egiten da
    1 + aldiz(x, r) ≥ 1 + aldiz(x, azkena_kendu(r))

  Beraz azpikasu honetan propietatea bete egiten da.

- **x ≠ z**

  - ✓ aldiz(x, z:r) $=^{(\#3)}$
    = aldiz(x, r)

  - ✓ aldiz(x, azkena_kendu(z:r)) $=^{(\#6)}$
    = aldiz(x, z:azkena_kendu(r)) $=^{(\#3)}$
    = aldiz(x, azkena_kendu(r))

    Indukzio-hipotesiagatik honako hau betetzen da

    aldiz(x, r) ≥ aldiz(x, azkena_kendu(r))

  Beraz azpikasu honetan ere propietatea bete egiten da.

  Bi azpikasu posibleetan propietatea bete egiten denez, kasu orokorrean propietatea bete egiten da.

23)hond(alder(s)) = alder(azkena_kendu(s)) (2009ko iraila) -- #

a) ++:: ([t], [t]) → [t]

[] ++ s = s (#1)
(x:r) ++ s = x:(r ++ s) (#2)

b) alder:: ([t]) → [t]

alder([]) = [] (#3)
alder(x:r) = alder(r) ++ (x:[]) (#4)

c) hond:: ([t]) → [t]

hond([]) = error "Zerrenda hutsa" (#5)
hond(x:r) = r (#6)

d) azkena_kendu:: ([t]) → [t]

azkena_kendu([]) = error "Zerrenda hutsa" (#7)
azkena_kendu(x:r)
| hutsa_da(r) = [] (#8)
| otherwise = x: azkena_kendu(r) (#9)

e) Orain s zerrendaren gainean indukzioa aplikatuz eta s hutsa ez dela jakinda, honako propietate hau frogatuko da:

hond(alder(s)) = alder(azkena_kendu(s))

Oinarrizko kasua: s = z:[]

hond(alder(z:[])) = alder(azkena_kendu(z:[]))?

✓ hond(alder(z:[])) =(#4)
= hond(alder([]) ++ (z:[])) =(#3)
= hond([] ++ (z:[])) =(#1)
= hond(z:[]) =(#6)
= []

✓ alder(azkena_kendu(z:[])) =(#8)
= alder([]) =(#3)
= []

Beraz oinarrizko kasuan propietatea bete egiten da.

Kasu orokorra: s = z:r, r zerrenda hutsa ez dela jakinda

hond(alder(z:r)) = alder(azkena_kendu(z:r))?

Indukzio-hipotesia (i.h.): (r zerrendak propietatea bete egiten du)

hond(alder(r)) = alder(azkena_kendu(r))

Frogatu nahi denera itzuliz:

hond(alder(z:r)) = alder(azkena_kendu(z:r))?

✓ hond(alder(z:r)) =(#4)
= hond(alder(r) ++ (z:[])) =(r hutsa ez denez, Prop2 propietateagatik badakigu alder(r) zerrenda ez dela hutsa, izan ere, r eta alder(r) zerrendek luzera bera dute eta ondorioz Prop1 aplikatu diezaiokegu alder(r) ++ (z:[]) zerrendari)
= hond(alder(r)) ++ (z:[])

✓ alder(azkena_kendu(z:r)) =(#9)
= alder(z:azkena_kendu(r)) =(#4)
= alder(azkena_kendu(r)) ++ (z:[]) =(i.h.)
= hond(alder(r)) ++ (z:[])

Beraz alde bietan gauza bera lortzen da eta ondorioz kasu orokorrean propietatea bete egiten da.

24) *batu(s) = batu(alder(s)) (2010eko apirila #1) -- #*

a) ++:: ([t], [t]) → [t]

[] ++ ℓ = ℓ (#1)
(*a*:q) ++ ℓ = *a*:(q ++ ℓ) (#2)

b) batu:: ([Int]) → Int

batu([]) = 0 (#3)
batu(*a*:q) = *a* + batu(q) (#4)

c) alder:: ([t]) → [t]

alder([]) = [] (#5)
alder(*a*:q) = alder(q) ++ (*a*:[]) (#6)

d) s-ren gainean indukzioa aplikatuz honako propietate hau frogatuko da orain:

batu(s) = batu(alder(s))

Oinarrizko kasua edo kasu sinplea: s = []

batu([]) = batu(alder([]))?

- ✓ batu([]) =(#3)
  = 0

- ✓ batu(alder([])) =(#5)
  = batu([]) =(#3)
  = 0

Betetzen da. Berdintzaren alde biek balio bera dute.

Kasu orokorra edo induktiboa: s = x:r

batu(x:r) = batu(alder(x:r))?

Indukzio-hipotesia: (r zerrendak propietatea bete egiten du)

batu(r) = batu(alder(r))

Frogatu nahi dugunera itzuliz: batu(x:r) = batu(alder(x:r))?

✓ batu(x:r) =$^{(\#4)}$
= x + batu(r)

✓ batu(alder(x:r)) =$^{(\#6)}$
= batu(alder(r) ++ (x:[])) =$^{(Prop)}$
= batu(alder(r)) + batu(x:[]) =$^{(\#4)}$
= batu(alder(r)) + (x + batu([])) =$^{(\#3)}$
= batu(alder(r)) + (x + 0) =$^{(0\ elementu\ neutroa\ da\ batuketarentzat)}$
= batu(alder(r)) + x =$^{(i.h.)}$
= batu(r) + x =$^{(batuketa\ trukakorra\ da)}$
= x + batu(r)

Beraz propietatea bete egiten da, alde bietan emaitza bera lortu dugulako.

Urrats bakoitzean koloreak erabili dira lerroaren eskuineko ertzean zehazten den ekuazioa formularen zein zatiri aplikatu zaion garbiago ikusteko. Irakurgarritasunagatik bi kolore desberdin erabili dira baina kolore desberdinek ez dute inolako esanahi berezirik.

25)*inkr(alder(s)) = alder(inkr(s)) (2010eko apirila #2) -- #*

a) ++:: ([t], [t]) → [t]

[] ++ ℓ = ℓ (#1)
(*a*:q) ++ ℓ = *a*:(q ++ ℓ) (#2)

b) inkr:: ([Int]) → [Int]

inkr([]) = 0 (#3)
inkr(*a*:q) = (*a* + 1) : inkr(q) (#4)

c) alder:: ([t]) → [t]

alder([]) = [] (#5)
alder(*a*:q) = alder(q) ++ (*a*:[]) (#6)

d) s-ren gainean indukzioa aplikatuz honako propietate hau frogatuko da orain:

inkr(alder(s)) = alder(inkr(s))

Oinarrizko kasua edo kasu sinplea: s = []

inkr(alder([])) = alder(inkr([]))?

✓ inkr(alder([])) =(#5)
= inkr([]) =(#3)
= []

✓ alder(inkr([])) =(#3)
= alder([]) =(#5)
= []

Betetzen da. Berdintzaren alde biek balio bera dute.

Kasu orokorra edo induktiboa: s = x:r
inkr(alder(x:r)) = alder(inkr(x:r))?

Indukzio-hipotesia: (r zerrendak propietatea bete egiten du)

inkr(alder(r)) = alder(inkr(r))

Frogatu nahi dugunera itzuliz: batu(x:r) = batu(alder(x:r))?

✓ inkr(alder(x:r)) $=^{(\#6)}$
= inkr(alder(r) ++ (x:[])) $=^{(Prop)}$
= inkr(alder(r)) ++ inkr(x:[]) $=^{(\#4)}$
= inkr(alder(r)) ++ ((x + 1): inkr([])) $=^{(\#3)}$
= inkr(alder(r)) ++ ((x + 1): [])

✓ alder(inkr(x:r)) $=^{(\#4)}$
= alder((x + 1):inkr(r)) $=^{(\#6)}$
= alder(inkr(r)) ++ ((x + 1):[]) $=^{(i.h.)}$
= inkr(alder(r)) ++ ((x + 1): [])

Beraz propietatea bete egiten da, alde bietan emaitza bera lortu dugulako.

Urrats bakoitzean koloreak erabili dira lerroaren eskuineko ertzean zehazten den ekuazioa formularen zein zatiri aplikatu zaion garbiago ikusteko. Irakurgarritasunagatik bi kolore desberdin erabili dira baina kolore desberdinek ez dute inolako esanahi berezirik.

26)*aldiz(x, alder(s)) = aldiz(x, s) (2010eko ekaina) -- #*

a) ++:: ([t], [t]) → [t]

[] ++ ℓ = ℓ (#1)
(*a*:q) ++ ℓ = *a*:(q ++ ℓ) (#2)

b) aldiz:: (t, [t]) → Int

aldiz(b, []) = 0 (#3)
aldiz(b, a:q)
|b == a = 1 + aldiz(b, q) (#4)
|otherwise = aldiz(b, q) (#5)

c) alder:: ([t]) → [t]

alder([]) = [] (#6)
alder(*a*:q) = alder(q) ++ (*a*:[]) (#7)

d) Orain s zerrendaren gainean indukzioa aplikatuz, honako propietate hau frogatuko da:

aldiz(x, alder(s)) = aldiz(x, s)

Oinarrizko kasua: s = []

aldiz(x, alder([])) = aldiz(x, [])?

✓ aldiz(x, alder([])) $=^{(\#6)}$
= aldiz(x, []) $=^{(\#3)}$
= 0

✓ aldiz(x, []) $=^{(\#3)}$
= 0

0 = 0 da.

Beraz oinarrizko kasuan propietatea bete egiten da.

Kasu orokorra: s = z:r

aldiz(x, alder(z:r)) = aldiz(x, z:r)?

Indukzio-hipotesia: (x elementuak eta r zerrendak propietatea bete egiten dute)

Indukzio-hipotesia: (x elementuak eta r zerrendak propietatea bete egiten dute)

aldiz(x, alder(r)) = aldiz(x, r)

Frogatu nahi denera itzuliz:

aldiz(x, alder(z:r)) = aldiz(x, z:r)?

Bi azpikasu edo aukera daude: x = z izatea edo x ≠ z izatea

- **x = z**

  ✓ aldiz(x, alder(z:r)) $=^{(\#7)}$
  = aldiz(x, alder(r) ++ (z:[])) $=^{(Prop)}$
  = aldiz(x, alder(r)) + aldiz(x, z:[]) $=^{(\#4)}$
  = aldiz(x, alder(r)) + 1 + aldiz(x, []) $=^{(\#3)}$
  = aldiz(x, alder(r)) + 1 + 0 $=^{(0 \text{ neutro da + eragilearentzat})}$
  = aldiz(x, alder(r)) + 1 $=^{(+ \text{ trukakorra da})}$
  = 1 + aldiz(x, alder(r))

  ✓ aldiz(x, z:r) $=^{(\#4)}$
  = 1 + aldiz(x, r) $=^{(i.h.)}$
  = 1 + aldiz(x, alder (r))

Beraz azpikasu honetan propietatea bete egiten da.

- **x ≠ z**

  ✓ aldiz(x, alder(z:r)) $=^{(\#7)}$
  = aldiz(x, alder(r) ++ (z:[])) $=^{(Prop)}$
  = aldiz(x, alder(r)) + aldiz(x, z:[]) $=^{(\#5)}$
  = aldiz(x, alder(r)) + aldiz(x, []) $=^{(\#3)}$
  = aldiz(x, alder(r)) + 0 $=^{(0 \text{ neutro da + eragilearentzat})}$
  = aldiz(x, alder(r))

  ✓ aldiz(x, z:r) $=^{(\#5)}$
  = aldiz(x, r) $=^{(i.h.)}$
  = aldiz(x, alder (r))

Beraz azpikasu honetan ere propietatea bete egiten da.

Bi azpikasu posibleetan propietatea bete egiten denez, kasu orokorrean propietatea bete egiten da.

<u>Oharra</u>: koloreen bidez lerro bakoitzean eskuineko ertzean zehazten den ekuazioa edo propietatea zein zatiri aplikatu zaion adierazten da. Irakurgarritasuna hobetzeko kolore desberdinak erabili dira.

27)*inkr(s ++ r) = inkr(s) ++ inkr(r) (2010eko iraila) -- #*

a) ++:: ([t], [t]) → [t]

[] ++ ℓ = ℓ (#1)
(a:q) ++ ℓ = a:(q ++ ℓ) (#2)

b) inkr:: ([Int]) → [Int]

inkr([]) = [] (#3)
inkr(a:q) = (a + 1) : inkr(q) (#4)

c) Orain s zerrendaren gainean indukzioa aplikatuz, honako propietate hau frogatuko da:

inkr(s ++ r) = inkr(s) ++ inkr(r)

Oinarrizko kasua: s = []

inkr([] ++ r) = inkr([]) ++ inkr(r)?

✓ inkr([] ++ r) =(#1)
= inkr(r)

✓ inkr([]) ++ inkr(r) =(#3)
= [] ++ inkr(r) =(#1)
= inkr(r)

Kasu bietan gauza bera lortu da: inkr(r)

Beraz oinarrizko kasuan propietatea bete egiten da.

Kasu orokorra: s = z:w

inkr((z:w) ++ r) = inkr(z:w) ++ inkr(r)?

Indukzio-hipotesia: (w eta r zerrendek propietatea bete egiten dute)

inkr(s ++ r) = inkr(s) ++ inkr(r)

Frogatu nahi denera itzuliz:

inkr((z:w) ++ r) = inkr(z:w) ++ inkr(r)?

- ✓ inkr((z:w) ++ r) =(#2)
  = inkr(z:(w ++ r)) =(#4)
  = (z + 1) : inkr(w ++ r)

- ✓ inkr(z:w) ++ inkr(r) =(#4)
  = ((z + 1) : inkr(w)) ++ inkr(r) =(#2)
  = (z + 1) : (inkr(w) ++ inkr(r)) =(i.h.)
  = (z + 1) : inkr(w ++ r)

  Kasu bietan gauza bera lortzen da: (z + 1) : inkr(w ++ r)

Beraz propietatea bete egiten da.

Oharra: Propietatea frogatzerakoan koloreen bidez lerro bakoitzean eskuineko ertzean zehazten den ekuazioa zein zatiri aplikatu zaion adierazten da.
Irakurgarritasuna hobetzeko kolore desberdinak erabili dira.