

### 3. Gaia: Programen Egiaztapena

#### 5. Ariketa-orria: Programa errekurtsiboak

1. Formulatu indukzio-hipotesi bat Hurrengo dei errekurtsiboentzat.

1.1.  $F$  funtzioaren goiburukoa eta espezifikazioa:

```
function F(x: Integer) return z: Integer;  
Aurre  $\equiv \{ x > 0 \}$   
Post  $\equiv \{ z = \lfloor \log_2 x \rfloor \}$  - -  $x$ -ren 2 oinarriko logaritmoaren  
- - zati osoa da  $z$ 
```

Kasu inuktiboan egiten den deia:

```
w := F(x/2);
```

1.2.  $konb$  funtzioak  $\binom{m}{n}$  zenbaki kombinatorioa kalkulatu du (postbaldintzan  $\binom{m}{n}$  zenbakia faktorialak erabiliz adierazten da).

```
function konb(m,n: Integer) return z: Integer is  
Aurre  $\equiv \{ 1 \leq n \leq m \}$   
  if n = 1 then  
    z := m;  
  else  
    zlag := konb(m,n-1);  
    z := (zlag/n)*(m-n+1);  
  end if;  
Aurre  $\equiv \{ z = \frac{m!}{n!(m-n)!} \}$ 
```

2. Hurrengo programetan postbaldintza egoki bat asmatu.

2.1.  $biderka$  funtzioak bi zenbaki arrunten biderkadura kalkulatu du.

```
function biderka(x,y: Integer) return z: Integer is  
Aurre  $\equiv \{ x \geq 0 \wedge y \geq 0 \}$   
  if x = 0 or y = 0 then  
    z := 0;  
  elsif x = 1 then  
    z := y;  
  else  
    z := biderka(x/2,2*y);  
    if x mod 2 /= 0 then  
      z := z+y;  
    end if;  
  end if;  
Post  $\equiv \{ \text{_____} \}$ 
```

2.2. *div* funtzioak bi zenbaki arrunten zatidura eta hondarra kalkulatzeko.

```

function div(x,y: Integer) return z,h: Integer is
Aurre  $\equiv \{ x \geq 0 \wedge y > 0 \}$ 
  if x < y then
    z := 0;
    h := x;
  else
    (z,h) := div(x-y,y);
    z := z+1;
  end if;
Post  $\equiv \{ \text{_____} \}$ 

```

3. Hurrengo programa errekursiboetan kasu inuktiboen frogapena egin.

3.1. *dig\_hand* funtzioak zenbaki arrunt baten digitu handiena kalkulatzeko.

```

function dig_hand(x: Integer) return y: Integer is
Aurre  $\equiv \{ x \geq 0 \}$ 
  if x <= 9 then
    y := x;
  else
    w := dig_hand(x/10);
    if w > x mod 10 then
      y := w;
    else
      y := x mod 10;
    end if;
  end if;
Post  $\equiv \{ y = \max\{ \frac{x}{10^{i-1}} \bmod 10 \mid i \geq 1 \} \}$ 

```

3.2. *konb* funtzioak  $m$  eta  $n$ -ren konbinatoria kalkulatzeko. Kontuan izan errekurtsibitateak aukera ematen digula faktorialak kalkulatzeko sortzen diren zenbaki handien kalkulua ebitatzeko, tarteko balioak erabiliz.

```

function konb(m,n: Integer) return r: Integer is
Aurre  $\equiv \{ m \geq n \geq 0 \}$ 
  r1,r2: Integer;
  if m = n or n = 0 then
    r := 1;
  else
    r1 := konb(m-1,n);
    r2 := konb(m-1,n-1);
    r := r1+r2;
  end if;
Post  $\equiv \{ r = \frac{m!}{n!(m-n)!} \}$ 

```

4. Idatzi honako programa errekursibo hauen aurre-ondoetako espezifikazioa, eta zuzenak direla egiaztatu.

- 4.1.  $x$  eta  $b$  ( $b > 1$ ) zenbaki osoko positiboak emanda,  $\log(b, x)$  funtzioak  $b$  oinarriko  $x$ -ren logaritmoaren zati osoa kalkulatzeko du  
(Oharra:  $(\log_b x = z) \leftrightarrow (b^z = x)$ ).

```
function log(b,x: Integer) return y: Integer is  
  if x = 1 or x < b then  
    y := 0;  
  else  
    y := log(b,x/b);  
    y := y+1;  
  end if;
```

- 4.2.  $\exp$  funtzioak berredura kalkulatzeko du.<sup>1</sup>

```
function exp(x,y: Integer) return r: Integer is  
Aurre  $\equiv \{ x > 0 \wedge y \geq 0 \}$   
  m: Integer;  
  if y = 0 then  
    r := 1;  
  elsif even(y) then  
    m := exp(x,y/2);  
    r := m*m;  
  else  
    m := exp(x,y/2);  
    r := m*m*x;  
  end if;
```

---

<sup>1</sup> $even(y) \equiv y \bmod 2 = 0$ .