```
⊢ ∀n,m:ℕ.  (∃p:{ℤ × ℤ| let x,y = p in GCD(m;n;(x * m) + (y * n))})
|
BY (D 0 THENA Auto)
|
1. n: ℕ
⊢ ∀m:ℕ. (∃p:{ℤ × ℤ| let x,y = p in GCD(m;n;(x * m) + (y * n))})
|
BY (GeneralNatInd 1 THENA Auto)
|
2. ∀n1:ℕn. ∀m:ℕ.  (∃p:{ℤ × ℤ| let x,y = p in GCD(m;n1;(x * m) + (y * n1))})
⊢ ∀m:ℕ. (∃p:{ℤ × ℤ| let x,y = p in GCD(m;n;(x * m) + (y * n))})
|
BY (D 0 THENA Auto)
|
3. m: ℕ
⊢ ∃p:{ℤ × ℤ| let x,y = p
|             in GCD(m;n;(x * m) + (y * n))}
|
BY (Decide ⌈n = 0⌉· THENA Auto)
|\
| 4. n = 0
| ⊢ ∃p:{ℤ × ℤ| let x,y = p
| |             in GCD(m;n;(x * m) + (y * n))}
| |
1 BY (SqHypSubst 4 0 THENA Auto)
| |
| ⊢ ∃p:{ℤ × ℤ| let x,y = p
| |             in GCD(m;0;(x * m) + (y * 0))}
| |
1 BY (InstConcl [⌈<1, 0>⌉]· THENA Auto)
| |
| ⊢ let x,y = <1, 0>
| |  in GCD(m;0;(x * m) + (y * 0))
| |
1 BY Reduce 0
| |
| ⊢ GCD(m;0;(1 * m) + 0)
| |
1 BY (InstLemma ‘gcd_p_zero‘ [⌈m⌉]· THENA Auto)
| |
| 5. GCD(m;0;m)
| ⊢ GCD(m;0;(1 * m) + 0)
| |
1 BY Auto
 \
  4. ¬(n = 0)
  ⊢ ∃p:{ℤ × ℤ| let x,y = p
  |             in GCD(m;n;(x * m) + (y * n))}
  |
  BY (InstHyp [⌈m rem n⌉;⌈n⌉] 2· THENA Auto)
  |\
  | ⊢ (m rem n) < n
  | |
```

```
1 BY (InstLemma ‘rem_bounds_1‘ [⌈m⌉;⌈n⌉]· THENA Auto)
| |
| 5. (0 ≤ (m rem n)) ∧ ((m rem n) < n)
| ⊢ (m rem n) < n
| |
1 BY Auto
 \
  5. ∃p:{ℤ × ℤ| let x,y = p
                in GCD(n;m rem n;(x * n) + (y * (m rem n)))}
  ⊢ ∃p:{ℤ × ℤ| let x,y = p
  |             in GCD(m;n;(x * m) + (y * n))}
  |
  BY D 5
  |
  5. p: ℤ × ℤ
  [6]. let x,y = p
       in GCD(n;m rem n;(x * n) + (y * (m rem n)))
  ⊢ ∃p:{ℤ × ℤ| let x,y = p
  |             in GCD(m;n;(x * m) + (y * n))}
  |
  BY D 5
  |
  5. p1: ℤ
  6. p2: ℤ
  [7]. let x,y = <p1, p2>
       in GCD(n;m rem n;(x * n) + (y * (m rem n)))
  ⊢ ∃p:{ℤ × ℤ| let x,y = p
  |             in GCD(m;n;(x * m) + (y * n))}
  |
  BY Reduce 7
  |
  [7]. GCD(n;m rem n;(p1 * n) + (p2 * (m rem n)))
  ⊢ ∃p:{ℤ × ℤ| let x,y = p
  |             in GCD(m;n;(x * m) + (y * n))}
  |
  BY (InstLemma ‘rem_to_div‘ [⌈m⌉;⌈n⌉]· THENA Auto)
  |
  8. (m rem n) = (m - (m ÷ n) * n)
  ⊢ ∃p:{ℤ × ℤ| let x,y = p
  |             in GCD(m;n;(x * m) + (y * n))}
  |
  BY Assert ⌈((p1 * n) + (p2 * (m rem n))) = ((p2 * m) + ((p1 - p2 * (m ÷ n)) * n))⌉·
  |\
  | 7. GCD(n;m rem n;(p1 * n) + (p2 * (m rem n)))
  | ⊢ ((p1 * n) + (p2 * (m rem n))) = ((p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
  | |
  1 BY (SqHypSubst 8 0 THENA Auto)
  | |
  | ⊢ ((p1 * n) + (p2 * (m - (m ÷ n) * n))) = ((p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
  | |
  1 BY Auto
   \
    9. ((p1 * n) + (p2 * (m rem n))) = ((p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
    ⊢ ∃p:{ℤ × ℤ| let x,y = p
```

```
|              in GCD(m;n;(x * m) + (y * n))}
|
BY (SqHypSubst 9 7 THENA Auto)
|
[7]. GCD(n;m rem n;(p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
⊢ ∃p:{ℤ × ℤ| let x,y = p
|              in GCD(m;n;(x * m) + (y * n))}
|
BY (InstConcl [⌈<p2, p1 - p2 * (m ÷ n)>⌉]· THENA Auto)
|
7. GCD(n;m rem n;(p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
⊢ let x,y = <p2, p1 - p2 * (m ÷ n)>
|  in GCD(m;n;(x * m) + (y * n))
|
BY Reduce 0
|
⊢ GCD(m;n;(p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
|
BY (InstLemma ‘div_rem_gcd_anne‘ [⌈m⌉;⌈n⌉;⌈(p2 * m) + ((p1 - p2 * (m ÷ n)) * n)⌉]·
|   THENA Auto
|   )
|
10. GCD(m;n;(p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
    ⟺ GCD(n;m rem n;(p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
⊢ GCD(m;n;(p2 * m) + ((p1 - p2 * (m ÷ n)) * n))
|
BY Auto
```

Extract:

```
λn. letrec bezout(n) =
  λm. if n = 0 then <1, 0>
      else let p1,p2 = bezout (m rem n) n
        in <p2, p1 - p2 * (m ÷ n)>
  in bezout(n)
```