```
multiply_functionality_wrt_eqmod  THM

⊢  ∀m,a,a',b,b':ℤ.
|  (a = a' mod m) ⇒ (b = b' mod m) ⇒ ((a·b) = (a'·b') mod m)
| by Unfold ‘eqmod‘ 0
|
1. m : ℤ
2. a : ℤ
3. a' : ℤ
4. b : ℤ
5. b' : ℤ
6. m | a-a'
7. m | b-b'
⊢  m | a·b-a'·b'
| by (Using [‘c',b]
|     (FwdThru Thm* ∀a,b,c:ℤ. a | b ⇒ a | b·c    {divisor_of_mul}
|      [6]))
|    THEN
|    (Using [‘c',a']
|     (FwdThru Thm* ∀a,b,c:ℤ. a | b ⇒ a | b·c    {divisor_of_mul}
|      [7]))
|    THEN
|    (OnHyps [7;6] Thin)
|
6. m | (a-a')·b
7. m | (b-b')·a'
⊢  m | a·b-a'·b'
| by FwdThru
|    Thm* ∀a,b1,b2:ℤ. a | b1 ⇒ a | b2 ⇒ a | b1+b2
|    {divisor_of_sum}
|    [6;7]
|
8. m | (a-a')·b+(b-b')·a'
⊢  m | a·b-a'·b'  by OnMClauses [0;8] ArithSimp
END
```