```
theory Chapter1
  imports HOL−Algebra.Algebra
begin
```

It seems that one cannot use ♡ as variable. My disappointment is immeasurable and my day is ruined.

```
theorem problem1A:
  assumes group: group G
  assumes subgrp: subgroup H G
  assumes proper: H ⊂ carrier G
  assumes iso: G ≅ G(|carrier := H|)
  shows infinite (carrier G)
proof
  from iso obtain h where bij: bij-betw h (carrier G) H
    unfolding is-iso-def iso-def by auto

  assume finite: finite (carrier G)
  with proper have card H < card (carrier G)
    by (simp add: psubset-card-mono)
  moreover from finite and bij have card H = card (carrier G)
    using bij-betw-same-card by fastforce
  ultimately show False by auto
qed
```

Problem 1B asks for a special case of Lagrange's theorem, thus we avoid using the general variant.

```
theorem (in comm-group) problem1B:
  assumes finite: finite (carrier G)
  assumes closed: g ∈ carrier G
  shows g [^] order G = 1
proof −
  let ?f = λx. g ⊗ x
  have [simp]: ?f ‘ carrier G = carrier G
    by (simp add: closed group.surj-const-mult)
  have inj-on ?f (carrier G)
    by (simp add: closed group.inj-on-cmult)
  hence (⨂x ∈ carrier G. x) = (⨂x ∈ carrier G. g ⊗ x)
   using finprod-reindex[where h=?f and A=carrier G and f=λx. x, symmetric]
    by simp
  also have ... = (⨂x ∈ carrier G. g) ⊗ (⨂x ∈ carrier G. x)
    using closed by (intro finprod-multf) auto
  finally have (⨂x ∈ carrier G. g) = 1
    using closed by (intro r-cancel-one'[THEN iffD1]) auto
  thus ?thesis
    using closed unfolding order-def by (simp add: finprod-const)
qed

end
```