```
Require Import Unicode.Utf8.

Module No1.
Import Unicode.Utf8.
  (*We first give the axioms of Principia
for the propositional calculus in *1.*)

Axiom MP1_1 : ∀  P Q : Prop,
  (P → Q) → P → Q. (*Modus ponens*)

  (**1.11 ommitted: it is MP for propositions containing variables. Likewi
se, ommitted the well-formedness rules 1.7, 1.71, 1.72*)

Axiom Taut1_2 : ∀ P : Prop, P ∨ P→ P. (*Tautology*)

Axiom Add1_3 : ∀ P Q : Prop, Q → P ∨ Q. (*Addition*)

Axiom Perm1_4 : ∀ P Q : Prop, P ∨ Q → Q ∨ P. (*Permutation*)

Axiom Assoc1_5 : ∀ P Q R : Prop, P ∨ (Q ∨ R) → Q ∨ (P ∨ R).

Axiom Sum1_6: ∀ P Q R : Prop, (Q → R) → (P ∨ Q → P ∨ R).
  (*These are all the propositional axioms of Principia Mathematica.*)

Axiom Impl1_01 : ∀ P Q : Prop, (P → Q) = (~P ∨ Q).
    (*This is a definition in Principia: there → is a defined sign and ∨, ~ are
primitive ones. The purposes of giving this as an Axiom are two: first, to
allow for the use of definitions in proofs, and second, to circumvent Coq's
 definitions of these primitive notions in Coq.*)

End No1.
```

```
Module No2.
Import No1.

(*We proceed to the deductions of *2 of Principia.*)

Theorem Abs2_01 : ∀ P : Prop,
  (P → ~P) → ~P.
Proof. intros P.
  specialize Taut1_2 with (~P).
  replace (~P ∨ ~P) with (P → ~P).
  apply MP1_1.
  apply Impl1_01.
Qed.

Theorem n2_02 : ∀ P Q : Prop,
  Q → (P → Q).
Proof. intros P Q.
  specialize Add1_3 with (~P) Q.
  replace (~P ∨ Q) with (P → Q).
  apply (MP1_1 Q (P → Q)).
  apply Impl1_01.
Qed.

Theorem n2_03 : ∀ P Q : Prop,
  (P → ~Q) → (Q → ~P).
Proof. intros P Q.
  specialize Perm1_4 with (~P) (~Q).
  replace (~P ∨ ~Q) with (P → ~Q).
  replace (~Q ∨ ~P) with (Q → ~P).
  apply (MP1_1 (P → ~Q) (Q → ~P)).
  apply Impl1_01.
  apply Impl1_01.
```

```
Qed.

Theorem Comm2_04 : ∀ P Q R : Prop,
 (P → (Q → R)) → (Q → (P → R)).
Proof. intros P Q R.
 specialize Assoc1_5 with (~P) (~Q) R.
 replace (~Q ∨ R) with (Q → R).
 replace (~P ∨ (Q → R)) with (P → (Q → R)).
 replace (~P ∨ R) with (P → R).
 replace (~Q ∨ (P → R)) with (Q → (P → R)).
 apply (MP1_1 (P → Q → R) (Q → P → R)).
 apply Impl1_01. apply Impl1_01.
 apply Impl1_01. apply Impl1_01.
Qed.

Theorem Syll2_05 : ∀ P Q R : Prop,
 (Q → R) → ((P  → Q) → (P → R)).
Proof. intros P Q R.
 specialize Sum1_6 with (~P) Q R.
 replace (~P ∨ Q) with (P → Q).
 replace (~P ∨ R) with (P → R).
 apply (MP1_1 (Q → R) ((P → Q) → (P → R))).
 apply Impl1_01. apply Impl1_01.
Qed.

Theorem Syll2_06 : ∀ P Q R : Prop,
 (P → Q) → ((Q → R) → (P → R)).
Proof. intros P Q R.
 specialize Comm2_04 with (Q → R) (P → Q) (P → R).
 intros Comm2_04.
 specialize Syll2_05 with P Q R. intros Syll2_05.
 specialize MP1_1 with
```

```
    ((Q → R) → (P → Q) → P → R) ((P → Q) → ((Q → R) → (P → R))).
 intros MP1_1.
 apply MP1_1.
 apply Comm2_04.
 apply Syll2_05.
Qed.

Theorem n2_07 : ∀ P : Prop,
 P → (P ∨ P).
Proof. intros P.
 specialize Add1_3 with P P.
 apply MP1_1.
Qed.

Theorem n2_08 : ∀ P : Prop,
 P → P.
Proof. intros P.
 specialize Syll2_05 with P (P ∨ P) P. intros Syll2_05.
 specialize Taut1_2 with P. intros Taut1_2.
 specialize MP1_1 with ((P ∨ P) → P) (P → P). intros MP1_1.
 apply Syll2_05.
 apply Taut1_2.
 apply n2_07.
Qed.

Theorem n2_1 : ∀ P : Prop,
 (~P) ∨ P.
Proof. intros P.
 specialize n2_08 with P.
 replace (~P ∨ P) with (P → P).
 apply MP1_1.
 apply Impl1_01.
```

```
Qed.

Theorem n2_11 : ∀ P : Prop,
  P ∨ ~P.
Proof. intros P.
  specialize Perm1_4 with (~P) P. intros Perm1_4.
  specialize n2_1 with P. intros Abs2_01.
  apply Perm1_4.
  apply n2_1.
Qed.

Theorem n2_12 : ∀ P : Prop,
  P → ~~P.
Proof. intros P.
  specialize n2_11 with (~P). intros n2_11.
  rewrite Impl1_01. assumption.
Qed.

Theorem n2_13 : ∀ P : Prop,
  P ∨ ~~~P.
Proof. intros P.
  specialize Sum1_6 with P (~P) (~~~P). intros Sum1_6.
  specialize n2_12 with (~P). intros n2_12.
  apply Sum1_6.
  apply n2_12.
  apply n2_11.
Qed.

Theorem n2_14 : ∀ P : Prop,
  ~~P → P.
Proof. intros P.
  specialize Perm1_4 with P (~~~P). intros Perm1_4.
```

```
  specialize n2_13 with P. intros n2_13.
  rewrite Impl1_01.
  apply Perm1_4.
  apply n2_13.
Qed.

Theorem Trans2_15 : ∀ P Q : Prop,
  (~P → Q) → (~Q → P).
Proof. intros P Q.
  specialize Syll2_05 with (~P) Q (~~Q). intros Syll2_05a.
  specialize n2_12 with Q. intros n2_12.
  specialize n2_03 with (~P) (~Q). intros n2_03.
  specialize Syll2_05 with (~Q) (~~P) P. intros Syll2_05b.
  specialize Syll2_05 with (~P → Q) (~P → ~~Q) (~Q → ~~P).
  intros Syll2_05c.
  specialize Syll2_05 with (~P → Q) (~Q → ~~P) (~Q → P).
  intros Syll2_05d.
  apply Syll2_05d.
  apply Syll2_05b.
  apply n2_14.
  apply Syll2_05c.
  apply n2_03.
  apply Syll2_05a.
  apply n2_12.
Qed.

Ltac Syll H1 H2 S :=
  let S := fresh S in match goal with
    | [ H1 : ?P -> ?Q, H2 : ?Q -> ?R |- _ ] =>
       assert (S : P -> R) by (intros p; apply (H2 (H1 p)))
end.
```

```
Ltac MP H1 H2 :=
  match goal with
    | [ H1 : ?P -> ?Q, H2 : ?P |- _ ] => specialize (H1 H2)
end.

Theorem Trans2_16 : ∀ P Q : Prop,
  (P → Q) → (~Q → ~P).
Proof. intros P Q.
  specialize n2_12 with Q. intros n2_12a.
  specialize Syll2_05 with P Q (~~Q). intros Syll2_05a.
  specialize n2_03 with P (~Q). intros n2_03a.
  MP n2_12a Syll2_05a.
  Syll Syll2_05a n2_03a S.
  apply S.
Qed.

Theorem Trans2_17 : ∀ P Q : Prop,
  (~Q → ~P) → (P → Q).
Proof. intros P Q.
  specialize n2_03 with (~Q) P. intros n2_03a.
  specialize n2_14 with Q. intros n2_14a.
  specialize Syll2_05 with P (~~Q) Q. intros Syll2_05a.
  MP n2_14a Syll2_05a.
  Syll n2_03a Syll2_05a S.
  apply S.
Qed.

Theorem n2_18 : ∀ P : Prop,
  (~P → P) → P.
Proof. intros P.
  specialize n2_12 with P. intro n2_12a.
  specialize Syll2_05 with (~P) P (~~P). intro Syll2_05a.
```

```
  MP Syll2_05a n2_12.
  specialize Abs2_01 with (~P). intros Abs2_01a.
  Syll Syll2_05a Abs2_01a Sa.
  specialize n2_14 with P. intros n2_14a.
  Syll H n2_14a Sb.
  apply Sb.
Qed.

Theorem n2_2 : ∀ P Q : Prop,
  P → (P ∨ Q).
Proof. intros P Q.
  specialize Add1_3 with Q P. intros Add1_3a.
  specialize Perm1_4 with Q P. intros Perm1_4a.
  Syll Add1_3a Perm1_4a S.
  apply S.
Qed.

Theorem n2_21 : ∀ P Q : Prop,
  ~P → (P → Q).
Proof. intros P Q.
  specialize n2_2 with (~P) Q. intros n2_2a.
  specialize Impl1_01 with P Q. intros Impl1_01a.
  replace (~P∨Q) with (P→Q) in n2_2a.
  apply n2_2a.
Qed.

Theorem n2_24 : ∀ P Q : Prop,
  P → (~P → Q).
Proof. intros P Q.
  specialize n2_21 with P Q. intros n2_21a.
  specialize Comm2_04 with (~P) P Q. intros Comm2_04a.
  apply Comm2_04a.
```

```
  apply n2_21a.
Qed.

Theorem n2_25 : ∀ P Q : Prop,
  P ∨ ((P ∨ Q) → Q).
Proof. intros P Q.
  specialize n2_1 with (P ∨ Q). intros n2_1a.
  specialize Assoc1_5 with (~(P∨Q)) P Q. intros Assoc1_5a.
  MP Assoc1_5a n2_1a.
  replace (~(P∨Q)∨Q) with (P∨Q→Q) in Assoc1_5a.
  apply Assoc1_5a.
  apply Impl1_01.
Qed.

Theorem n2_26 : ∀ P Q : Prop,
  ~P ∨ ((P → Q) → Q).
Proof. intros P Q.
  specialize n2_25 with (~P) Q. intros n2_25a.
  replace (~P∨Q) with (P→Q) in n2_25a.
  apply n2_25a.
  apply Impl1_01.
Qed.

Theorem n2_27 : ∀ P Q : Prop,
  P → ((P → Q) → Q).
Proof. intros P Q.
  specialize n2_26 with P Q. intros n2_26a.
  replace (~P∨((P→Q)→Q)) with (P→(P→Q)→Q) in n2_26a.
  apply n2_26a.
  apply Impl1_01.
Qed.
```

```
Theorem n2_3 : ∀ P Q R : Prop,
  (P ∨ (Q ∨ R)) → (P ∨ (R ∨ Q)).
Proof. intros P Q R.
  specialize Perm1_4 with Q R. intros Perm1_4a.
  specialize Sum1_6 with P (Q∨R) (R∨Q). intros Sum1_6a.
  MP Sum1_6a Perm1_4a.
  apply Sum1_6a.
Qed.

Theorem n2_31 : ∀ P Q R : Prop,
  (P ∨ (Q ∨ R)) → ((P ∨ Q) ∨ R).
Proof. intros P Q R.
  specialize n2_3 with P Q R. intros n2_3a.
  specialize Assoc1_5 with P R Q. intros Assoc1_5a.
  specialize Perm1_4 with R (P∨Q). intros Perm1_4a.
  Syll Assoc1_5a Perm1_4a Sa.
  Syll n2_3a Sa Sb.
  apply Sb.
Qed.

Theorem n2_32 : ∀ P Q R : Prop,
  ((P ∨ Q) ∨ R) → (P ∨ (Q ∨ R)).
Proof. intros P Q R.
  specialize Perm1_4 with (P∨Q) R. intros Perm1_4a.
  specialize Assoc1_5 with R P Q. intros Assoc1_5a.
  specialize n2_3 with P R Q. intros n2_3a.
  specialize Syll2_06 with ((P∨Q)∨R) (R∨P∨Q) (P∨R∨Q).
  intros Syll2_06a.
  MP Syll2_06a Perm1_4a.
  MP Syll2_06a Assoc1_5a.
  specialize Syll2_06 with ((P∨Q)∨R) (P∨R∨Q) (P∨Q∨R).
  intros Syll2_06b.
```

```
  MP Syll2_06b Syll2_06a.
  MP Syll2_06b n2_3a.
  apply Syll2_06b.
Qed.

(* Axiom n2_33 : ∀ P Q R : Prop,
  (P∨Q∨R)=((P∨Q)∨R)
  This definition makes the default left association.*)

Theorem n2_36 : ∀ P Q R : Prop,
  (Q → R) → ((P ∨ Q) → (R ∨ P)).
Proof. intros P Q R.
  specialize Perm1_4 with P R. intros Perm1_4a.
  specialize Syll2_05 with (P∨Q) (P∨R) (R∨P). intros Syll2_05a.
  MP Syll2_05a Perm1_4a.
  specialize Sum1_6 with P Q R. intros Sum1_6a.
  Syll Sum1_6a Syll2_05a S.
  apply S.
Qed.

Theorem n2_37 : ∀ P Q R : Prop,
  (Q → R) → ((Q ∨ P) → (P ∨ R)).
Proof. intros P Q R.
  specialize Perm1_4 with Q P. intros Perm1_4a.
  specialize Syll2_06 with (Q∨P) (P∨Q) (P∨R). intros Syll2_06a.
  MP Syll2_05a Perm1_4a.
  specialize Sum1_6 with P Q R. intros Sum1_6a.
  Syll Sum1_6a Syll2_05a S.
  apply S.
Qed.

Theorem n2_38 : ∀ P Q R : Prop,
```

```
  (Q → R) → ((Q ∨ P) → (R ∨ P)).
Proof. intros P Q R.
  specialize Perm1_4 with P R. intros Perm1_4a.
  specialize Syll2_05 with (Q∨P) (P∨R) (R∨P). intros Syll2_05a.
  MP Syll2_05a Perm1_4a.
  specialize Perm1_4 with Q P. intros Perm1_4b.
  specialize Syll2_06 with (Q∨P) (P∨Q) (P∨R). intros Syll2_06a.
  MP Syll2_06a Perm1_4b.
  Syll Syll2_06a Syll2_05a H.
  specialize Sum1_6 with P Q R. intros Sum1_6a.
  Syll Sum1_6a H S.
  apply S.
Qed.

Theorem n2_4 : ∀ P Q : Prop,
  (P ∨ (P ∨ Q)) → (P ∨ Q).
Proof. intros P Q.
  specialize n2_31 with P P Q. intros n2_31a.
  specialize Taut1_2 with P. intros Taut1_2a.
  specialize n2_38 with Q (P∨P) P. intros n2_38a.
  MP n2_38a Taut1_2a.
  Syll n2_31a n2_38a S.
  apply S.
Qed.

Theorem n2_41 : ∀ P Q : Prop,
  (Q ∨ (P ∨ Q)) → (P ∨ Q).
Proof. intros P Q.
  specialize Assoc1_5 with Q P Q. intros Assoc1_5a.
  specialize Taut1_2 with Q. intros Taut1_2a.
  specialize Sum1_6 with P (Q∨Q) Q. intros Sum1_6a.
  MP Sum1_6a Taut1_2a.
```

```
 Syll Assoc1_5a Sum1_6a S.
 apply S.
Qed.

Theorem n2_42 : ∀ P Q : Prop,
 (~P ∨ (P → Q)) → (P → Q).
Proof. intros P Q.
 specialize n2_4 with (~P) Q. intros n2_4a.
 replace (~P∨Q) with (P→Q) in n2_4a.
 apply n2_4a. apply Impl1_01.
Qed.

Theorem n2_43 : ∀ P Q : Prop,
 (P → (P → Q)) → (P → Q).
Proof. intros P Q.
 specialize n2_42 with P Q. intros n2_42a.
 replace (~P ∨ (P→Q)) with (P→(P→Q)) in n2_42a.
 apply n2_42a. apply Impl1_01.
Qed.

Theorem n2_45 : ∀ P Q : Prop,
 ~(P ∨ Q) → ~P.
Proof. intros P Q.
 specialize n2_2 with P Q. intros n2_2a.
 specialize Trans2_16 with P (P∨Q). intros Trans2_16a.
 MP n2_2 Trans2_16a.
 apply Trans2_16a.
Qed.

Theorem n2_46 : ∀ P Q : Prop,
 ~(P ∨ Q) → ~Q.
Proof. intros P Q.
```

```
  specialize Add1_3 with P Q. intros Add1_3a.
  specialize Trans2_16 with Q (P∨Q). intros Trans2_16a.
  MP Add1_3a Trans2_16a.
  apply Trans2_16a.
Qed.

Theorem n2_47 : ∀ P Q : Prop,
  ~(P ∨ Q) → (~P ∨ Q).
Proof. intros P Q.
  specialize n2_45 with P Q. intros n2_45a.
  specialize n2_2 with (~P) Q. intros n2_2a.
  Syll n2_45a n2_2a S.
  apply S.
Qed.

Theorem n2_48 : ∀ P Q : Prop,
  ~(P ∨ Q) → (P ∨ ~Q).
Proof. intros P Q.
  specialize n2_46 with P Q. intros n2_46a.
  specialize Add1_3 with P (~Q). intros Add1_3a.
  Syll n2_46a Add1_3a S.
  apply S.
Qed.

Theorem n2_49 : ∀ P Q : Prop,
  ~(P ∨ Q) → (~P ∨ ~Q).
Proof. intros P Q.
  specialize n2_45 with P Q. intros n2_45a.
  specialize n2_2 with (~P) (~Q). intros n2_2a.
  Syll n2_45a n2_2a S.
  apply S.
Qed.
```

```
Theorem n2_5 : ∀ P Q : Prop,
  ~(P → Q) → (~P → Q).
Proof. intros P Q.
  specialize n2_47 with (~P) Q. intros n2_47a.
  replace (~P∨Q) with (P→Q) in n2_47a.
  replace (~~P∨Q) with (~P→Q) in n2_47a.
  apply n2_47a.
  apply Impl1_01. apply Impl1_01.
Qed.

Theorem n2_51 : ∀ P Q : Prop,
  ~(P → Q) → (P → ~Q).
Proof. intros P Q.
  specialize n2_48 with (~P) Q. intros n2_48a.
  replace (~P∨Q) with (P→Q) in n2_48a.
  replace (~P∨~Q) with (P→~Q) in n2_48a.
  apply n2_48a.
  apply Impl1_01. apply Impl1_01.
Qed.

Theorem n2_52 : ∀ P Q : Prop,
  ~(P → Q) → (~P → ~Q).
Proof. intros P Q.
  specialize n2_49 with (~P) Q. intros n2_49a.
  replace (~P∨Q) with (P→Q) in n2_49a.
  replace (~~P∨~Q) with (~P→~Q) in n2_49a.
  apply n2_49a.
  apply Impl1_01. apply Impl1_01.
Qed.

Theorem n2_521 : ∀ P Q : Prop,
```

```
 ~(P→Q)→(Q→P).
Proof. intros P Q.
 specialize n2_52 with P Q. intros n2_52a.
 specialize Trans2_17 with Q P. intros Trans2_17a.
 Syll n2_52a Trans2_17a S.
 apply S.
Qed.

Theorem n2_53 : ∀ P Q : Prop,
 (P ∨ Q) → (~P -> Q).
Proof. intros P Q.
 specialize n2_12 with P. intros n2_12a.
 specialize n2_38 with Q P (~~P). intros n2_38a.
 MP n2_38a n2_12a.
 replace (~~P∨Q) with (~P→Q) in n2_38a.
 apply n2_38a. apply Impl1_01.
Qed.

Theorem n2_54 : ∀ P Q : Prop,
 (~P → Q) → (P ∨ Q).
Proof. intros P Q.
 specialize n2_14 with P. intros n2_14a.
 specialize n2_38 with Q (~~P) P. intros n2_38a.
 MP n2_38a n2_12a.
 replace (~~P∨Q) with (~P→Q) in n2_38a.
 apply n2_38a. apply Impl1_01.
Qed.

Theorem n2_55 : ∀ P Q : Prop,
 ~P → ((P ∨ Q) → Q).
Proof. intros P Q.
 specialize n2_53 with P Q. intros n2_53a.
```

```
  specialize Comm2_04 with (P∨Q) (~P) Q. intros Comm2_04a.
  MP n2_53a Comm2_04a.
  apply Comm2_04a.
Qed.

Theorem n2_56 : ∀ P Q : Prop,
  ~Q → ((P ∨ Q) → P).
Proof. intros P Q.
  specialize n2_55 with Q P. intros n2_55a.
  specialize Perm1_4 with P Q. intros Perm1_4a.
  specialize Syll2_06 with (P∨Q) (Q∨P) P. intros Syll2_06a.
  MP Syll2_06a Perm1_4a.
  Syll n2_55a Syll2_06a S.
  apply S.
Qed.

Theorem n2_6 : ∀ P Q : Prop,
  (~P→Q) → ((P → Q) → Q).
Proof. intros P Q.
  specialize n2_38 with Q (~P) Q. intros n2_38a.
  specialize Taut1_2 with Q. intros Taut1_2a.
  specialize Syll2_05 with (~P∨Q) (Q∨Q) Q. intros Syll2_05a.
  MP Syll2_05a Taut1_2a.
  Syll n2_38a Syll2_05a S.
  replace (~P∨Q) with (P→Q) in S.
  apply S.
  apply Impl1_01.
Qed.

Theorem n2_61 : ∀ P Q : Prop,
  (P → Q) → ((~P → Q) → Q).
Proof. intros P Q.
```

```
  specialize n2_6 with P Q. intros n2_6a.
  specialize Comm2_04 with (~P→Q) (P→Q) Q. intros Comm2_04a.
  MP Comm2_04a n2_6a.
  apply Comm2_04a.
Qed.

Theorem n2_62 : ∀ P Q : Prop,
  (P ∨ Q) → ((P → Q) → Q).
Proof. intros P Q.
  specialize n2_53 with P Q. intros n2_53a.
  specialize n2_6 with P Q. intros n2_6a.
  Syll n2_53a n2_6a S.
  apply S.
Qed.

Theorem n2_621 : ∀ P Q : Prop,
  (P → Q) → ((P ∨ Q) → Q).
Proof. intros P Q.
  specialize n2_62 with P Q. intros n2_62a.
  specialize Comm2_04 with (P ∨ Q) (P→Q) Q. intros Comm2_04a.
  MP Comm2_04a n2_62a. apply Comm2_04a.
Qed.

Theorem n2_63 : ∀ P Q : Prop,
  (P ∨ Q) → ((~P ∨ Q) → Q).
Proof. intros P Q.
  specialize n2_62 with P Q. intros n2_62a.
  replace (~P∨Q) with (P→Q).
  apply n2_62a.
  apply Impl1_01.
Qed.
```

```
Theorem n2_64 : ∀ P Q : Prop,
 (P ∨ Q) → ((P ∨ ~Q) → P).
Proof. intros P Q.
 specialize n2_63 with Q P. intros n2_63a.
 specialize Perm1_4 with P Q. intros Perm1_4a.
 Syll n2_63a Perm1_4a Ha.
 specialize Syll2_06 with (P∨~Q) (~Q∨P) P. intros Syll2_06a.
 specialize Perm1_4 with P (~Q). intros Perm1_4b.
 MP Syll2_05a Perm1_4b.
 Syll Syll2_05a Ha S.
 apply S.
Qed.

Theorem n2_65 : ∀ P Q : Prop,
 (P → Q) → ((P → ~Q) → ~P).
Proof. intros P Q.
 specialize n2_64 with (~P) Q. intros n2_64a.
 replace (~P∨Q) with (P→Q) in n2_64a.
 replace (~P∨~Q) with (P→~Q) in n2_64a.
 apply n2_64a.
 apply Impl1_01. apply Impl1_01.
Qed.

Theorem n2_67 : ∀ P Q : Prop,
 ((P ∨ Q) → Q) → (P → Q).
Proof. intros P Q.
 specialize n2_54 with P Q. intros n2_54a.
 specialize Syll2_06 with (~P→Q) (P∨Q) Q. intros Syll2_06a.
 MP Syll2_06a n2_54a.
 specialize n2_24 with  P Q. intros n2_24.
 specialize Syll2_06 with P (~P→Q) Q. intros Syll2_06b.
 MP Syll2_06b n2_24a.
```

```
  Syll Syll2_06b Syll2_06a S.
  apply S.
Qed.

Theorem n2_68 : ∀ P Q : Prop,
  ((P → Q) → Q) → (P ∨ Q).
Proof. intros P Q.
  specialize n2_67 with (~P) Q. intros n2_67a.
  replace (~P∨Q) with (P→Q) in n2_67a.
  specialize n2_54 with P Q. intros n2_54a.
  Syll n2_67a n2_54a S.
  apply S.
  apply Impl1_01.
Qed.

Theorem n2_69 : ∀ P Q : Prop,
  ((P → Q) → Q) → ((Q → P) → P).
Proof. intros P Q.
  specialize n2_68 with P Q. intros n2_68a.
  specialize Perm1_4 with P Q. intros Perm1_4a.
  Syll n2_68a Perm1_4a Sa.
  specialize n2_62 with Q P. intros n2_62a.
  Syll Sa n2_62a Sb.
  apply Sb.
Qed.

Theorem n2_73 : ∀ P Q R : Prop,
  (P → Q) → (((P ∨ Q) ∨ R) → (Q ∨ R)).
Proof. intros P Q R.
  specialize n2_621 with P Q. intros n2_621a.
  specialize n2_38 with R (P∨Q) Q. intros n2_38a.
  Syll n2_621a n2_38a S.
```

```
  apply S.
Qed.

Theorem n2_74 : ∀ P Q R : Prop,
  (Q → P) → ((P ∨ Q) ∨ R) → (P ∨ R).
Proof. intros P Q R.
  specialize n2_73 with Q P R. intros n2_73a.
  specialize Assoc1_5 with P Q R. intros Assoc1_5a.
  specialize n2_31 with Q P R. intros n2_31a. (*not cited explicitly!*)
  Syll Assoc1_5a n2_31a Sa.
  specialize n2_32 with P Q R. intros n2_32a. (*not cited explicitly!*)
  Syll n2_32a Sa Sb.
  specialize Syll2_06 with ((P∨Q)∨R) ((Q∨P)∨R) (P∨R).
  intros Syll2_06a.
  MP Syll2_06a Sb.
  Syll n2_73a Syll2_05a H.
  apply H.
Qed.

Theorem n2_75 : ∀ P Q R : Prop,
  (P ∨ Q) → ((P ∨ (Q → R)) → (P ∨ R)).
Proof. intros P Q R.
  specialize n2_74 with P (~Q) R. intros n2_74a.
  specialize n2_53 with Q P. intros n2_53a.
  Syll n2_53a n2_74a Sa.
  specialize n2_31 with P (~Q) R. intros n2_31a.
  specialize Syll2_06 with (P∨(~Q)∨R)((P∨(~Q))∨R)  (P∨R).
  intros Syll2_06a.
  MP Syll2_06a n2_31a.
  Syll Sa Syll2_06a Sb.
  specialize Perm1_4 with P Q.
  intros Perm1_4a. (*not cited!*)
```

```
  Syll Perm1_4a Sb Sc.
  replace (~Q∨R) with (Q→R) in Sc.
  apply Sc.
  apply Impl1_01.
Qed.

Theorem n2_76 : ∀ P Q R : Prop,
  (P ∨ (Q → R)) → ((P ∨ Q) → (P ∨ R)).
Proof. intros P Q R.
  specialize n2_75 with P Q R. intros n2_75a.
  specialize Comm2_04 with (P∨Q) (P∨(Q→R)) (P∨R).
  intros Comm2_04a.
  MP Comm2_04a n2_75a.
  apply Comm2_04a.
Qed.

Theorem n2_77 : ∀ P Q R : Prop,
  (P → (Q → R)) → ((P → Q) → (P → R)).
Proof. intros P Q R.
  specialize n2_76 with (~P) Q R. intros n2_76a.
  replace (~P∨(Q→R)) with (P→Q→R) in n2_76a.
  replace (~P∨Q) with (P→Q) in n2_76a.
  replace (~P∨R) with (P→R) in n2_76a.
  apply n2_76a.
  apply Impl1_01. apply Impl1_01. apply Impl1_01.
Qed.

Theorem n2_8 : ∀ Q R S : Prop,
  (Q ∨ R) → ((~R ∨ S) → (Q ∨ S)).
Proof. intros Q R S.
  specialize n2_53 with R Q. intros n2_53a.
  specialize Perm1_4 with Q R. intros Perm1_4a.
```

```
  Syll Perm1_4a n2_53a Ha.
  specialize n2_38 with S (~R) Q. intros n2_38a.
  Syll H n2_38a Hb.
  apply Hb.
Qed.

Theorem n2_81 : ∀ P Q R S : Prop,
  (Q → (R → S)) → ((P ∨ Q) → ((P ∨ R) → (P ∨ S))).
Proof. intros P Q R S.
  specialize Sum1_6 with P Q (R→S). intros Sum1_6a.
  specialize n2_76 with P R S. intros n2_76a.
  specialize Syll2_05 with (P∨Q) (P∨(R→S)) ((P∨R)→(P∨S)).
  intros Syll2_05a.
  MP Syll2_05a n2_76a.
  Syll Sum1_6a Syll2_05a H.
  apply H.
Qed.

Theorem n2_82 : ∀ P Q R S : Prop,
  (P ∨ Q ∨ R)→((P ∨ ~R ∨ S)→(P ∨ Q ∨ S)).
Proof. intros P Q R S.
  specialize n2_8 with Q R S. intros n2_8a.
  specialize n2_81 with P (Q∨R) (~R∨S) (Q∨S). intros n2_81a.
  MP n2_81a n2_8a.
  apply n2_81a.
Qed.

Theorem n2_83 : ∀ P Q R S : Prop,
  (P→(Q→R))→((P→(R→S))→(P→(Q→S))).
Proof. intros P Q R S.
  specialize n2_82 with (~P) (~Q) R S. intros n2_82a.
  replace (~Q∨R) with (Q→R) in n2_82a.
```

```
  replace (~P∨(Q→R)) with (P→Q→R) in n2_82a.
  replace (~R∨S) with (R→S) in n2_82a.
  replace (~P∨(R→S)) with (P→R→S) in n2_82a.
  replace (~Q∨S) with (Q→S) in n2_82a.
  replace (~Q∨S) with (Q→S) in n2_82a.
  replace (~P∨(Q→S)) with (P→Q→S) in n2_82a.
  apply n2_82a.
  apply Impl1_01.
  apply Impl1_01.
  apply Impl1_01.
  apply Impl1_01.
  apply Impl1_01.
  apply Impl1_01.
  apply Impl1_01.
Qed.

Theorem n2_85 : ∀ P Q R : Prop,
  ((P ∨ Q) → (P ∨ R)) → (P ∨ (Q → R)).
Proof. intros P Q R.
  specialize Add1_3 with P Q. intros Add1_3a.
  specialize Syll2_06 with Q (P∨Q) R. intros Syll2_06a.
  MP Syll2_06a Add1_3a.
  specialize n2_55 with P R. intros n2_55a.
  specialize Syll2_05 with (P∨Q) (P∨R) R. intros Syll2_05a.
  Syll n2_55a Syll2_05a Ha.
  specialize n2_83 with (~P) ((P∨Q)→(P∨R)) ((P∨Q)→R) (Q→R). intros n
2_83a.
  MP n2_83a Ha.
  specialize Comm2_04 with (~P) (P∨Q→P∨R) (Q→R). intros Comm2_04a
.
  Syll Ha Comm2_04a Hb.
  specialize n2_54 with P (Q→R). intros n2_54a.
```

```
  specialize n2_02 with (~P) ((P∨Q→R)→(Q→R)). intros n2_02a.
(*Not mentioned! Greg's suggestion per the BRS list in June 25, 2017.*)
  MP Syll2_06a n2_02a.
  MP Hb n2_02a.
  Syll Hb n2_54a Hc.
  apply Hc.
Qed.

Theorem n2_86 : ∀ P Q R : Prop,
  ((P → Q) → (P → R)) → (P → (Q →  R)).
Proof. intros P Q R.
  specialize n2_85 with (~P) Q R. intros n2_85a.
  replace (~P∨Q) with (P→Q) in n2_85a.
  replace (~P∨R) with (P→R) in n2_85a.
  replace (~P∨(Q→R)) with (P→Q→R) in n2_85a.
  apply n2_85a.
  apply Impl1_01. apply Impl1_01. apply Impl1_01.
Qed.

End No2.
```