Prenex Form Transformation Rules Justified in JAPE

Robert Keller

September 2007

We want to show the rules for transforming formulas to prenex form and how they are justified in JAPE. Note that although the rules are applied only in one direction, they need to be justified as logical equivalences and not just implication.

Each rule has a symmetric case, where the two sides of the connective are flipped. For brevity, we only show one side of each rule. There are thus four rules to consider (where here $\Rightarrow$ means "replace with"). In each case, there is a requirement that x is not free in F. However, we normally would handle this by renaming variables beforehand, since any occurrence of x in F would not be in the scope of the quantifier on the left-hand side.

1. $(\forall x\ E) \wedge F \Rightarrow \forall x\ (E \wedge F)$, where x is not free in F
2. $(\forall x\ E) \vee F \Rightarrow \forall x\ (E \vee F)$ "
3. $(\exists x\ E) \wedge F \Rightarrow \exists x\ (E \wedge F)$ "
4. $(\exists x\ E) \vee F \Rightarrow \exists x\ (E \vee F)$ "

We give JAPE proofs that these are equivalent. In particular, we are interested in the way in which JAPE handles the proviso: It attaches the proviso "x not in F" when we try to derive the relevant sub-formula F[x\i] from F using the ***hyp*** rule.

∀x.E∧F ⊢ ∀x.(E∧F)	actual i, ∀x.(E∧F) ⊢ ∀x.E∧F
1: ∀x.E∧F premise 2: F ∧ elim 1 3: ∀x.E ∧ elim 1 4: actual i assumption 5: E[x\i] ∀ elim 3,4 6: E[x\i]∧F ∧ intro 5,2 7: ∀x.(E∧F) ∀ intro 4-6 Provided: x NOTIN F	1: actual i, ∀x.(E∧F) premises 2: actual i1 assumption 3: E[x\i1]∧F ∀ elim 1.2,2 4: E[x\i1] ∧ elim 3 5: ∀x.E ∀ intro 2-4 6: E[x\i]∧F ∀ elim 1.2,1.1 7: F ∧ elim 6 8: ∀x.E∧F ∧ intro 5,7 Provided: x NOTIN F

∀x.E∨F ⊢ ∀x.(E∨F)

1:	∀x.E∨F	premise
2:	∀x.E	assumption
3:	actual i	assumption
4:	E[x\i]	∀ elim 2,3
5:	E[x\i]∨F	∨ intro 4
6:	∀x.(E∨F)	∀ intro 3-5
7:	F	assumption
8:	E∨F	∨ intro 7
9:	actual i1	assumption
10:	E[x\i1]∨F	∨ intro 7
11:	∀x.(E∨F)	∀ intro 9-10
12:	∀x.(E∨F)	∨ elim 1,2-6,7-11

Provided:
x NOTIN F

∀x.(E∨F) ⊢ ∀x.E∨F

1:	∀x.(E∨F)	premise
2:	F∨¬F	Theorem E∨¬E
3:	F	assumption
4:	∀x.E∨F	∨ intro 3
5:	¬F	assumption
6:	actual i	assumption
7:	E[x\i]∨F	∀ elim 1,6
8:	E[x\i]	assumption
9:	F	assumption
10:	⊥	¬ elim 9,5
11:	E[x\i]	contra (constructive) 10
12:	E[x\i]	∨ elim 7,8-8,9-11
13:	∀x.E	∀ intro 6-12
14:	∀x.E∨F	∨ intro 13
15:	∀x.E∨F	∨ elim 2,3-4,5-14

Provided:
x NOTIN F

actual i, ∃x.E∨F ⊢ ∃x.(E∨F)

1:	actual i, ∃x.E∨F	premises
2:	∃x.E	assumption
3:	actual i1, E[x\i1]	assumptions
4:	E[x\i1]∨F	∨ intro 3.2
5:	∃x.(E∨F)	∃ intro 4,3.1
6:	∃x.(E∨F)	∃ elim 2,3-5
7:	F	assumption
8:	E[x\i]∨F	∨ intro 7
9:	∃x.(E∨F)	∃ intro 8,1.1
10:	∃x.(E∨F)	∨ elim 1.2,2-6,7-9

Provided:
x NOTIN F

∃x.(E∨F) ⊢ ∃x.E∨F

1:	∃x.(E∨F)	premise
2:	actual i, E[x\i]∨F	assumptions
3:	E[x\i]	assumption
4:	∃x.E	∃ intro 3,2.1
5:	∃x.E∨F	∨ intro 4
6:	F	assumption
7:	∃x.E∨F	∨ intro 6
8:	∃x.E∨F	∨ elim 2.2,3-5,6-7
9:	∃x.E∨F	∃ elim 1,2-8

Provided:
x NOTIN F

∃x.E∧F ⊢ ∃x.(E∧F)	
1: ∃x.E∧F	premise
2: F	∧ elim 1
3: ∃x.E	∧ elim 1
4: actual i, E[x\i]	assumptions
5: E[x\i]∧F	∧ intro 4.2,2
6: ∃x.(E∧F)	∃ intro 5,4.1
7: ∃x.(E∧F)	∃ elim 3,4-6

Provided:
x NOTIN F

actual i, ∃x.(E∧F) ⊢ ∃x.E∧F	
1: actual i, ∃x.(E∧F)	premises
2: actual i1, E[x\i1]∧F	assumptions
3: E[x\i1]	∧ elim 2.2
4: ∃x.E	∃ intro 3,2.1
5: ∃x.E	∃ elim 1.2,2-4
6: actual i2, E[x\i2]∧F	assumptions
7: F	∧ elim 6.2
8: F	∃ elim 1.2,6-7
9: ∃x.E∧F	∧ intro 5,8

Provided:
x NOTIN F