The paragraph about Superman asserts that its last sentence follows from the first four, so it can be written as the following expression:

$$F0 \wedge F1 \wedge F2 \wedge F3 \;\Rightarrow\; \neg e \quad .$$

The reason for giving a name (i.e. a boolean variable) to each sentence now becomes clear; had we used the sentences themselves instead of their names, the final expression would have been long and unwieldy. To determine the validity of the Superman paragraph, we have to see whether this expression is *true* in all states. Rather than do this (there are $2^6 = 64$ states to check!), let us wait until we have learned rules and methods for manipulating and simplifying expressions given in the next chapter.

Exercises for Chapter 2

2.1 Each line below contains an expression and two states *S0* and *S1* (using t for *true* and f for *false*). Evaluate the expression in both states.

		state *S0*				state *S1*			
	expression	m	n	p	q	m	n	p	q
(a)	$\neg(m \vee n)$	t	f	t	t	f	t	t	t
(b)	$\neg m \vee n$	t	f	t	t	f	t	t	t
(c)	$\neg(m \wedge n)$	t	f	t	t	f	t	t	t
(d)	$\neg m \wedge n$	t	f	t	t	f	t	t	t
(e)	$(m \vee n) \Rightarrow p$	t	f	t	t	t	t	f	t
(f)	$m \vee (n \Rightarrow p)$	t	f	t	t	t	t	f	t
(g)	$(m \equiv n) \wedge (p \equiv q)$	f	f	t	f	t	f	t	f
(h)	$(m \equiv (n \wedge (p \equiv q))$	f	f	t	f	t	f	t	f
(i)	$(m \equiv (n \wedge p \equiv q)$	f	f	t	f	t	f	t	f
(j)	$(m \equiv n) \wedge (p \Rightarrow q)$	f	t	f	t	t	t	f	f
(k)	$(m \equiv n \wedge p) \Rightarrow q$	f	t	f	t	t	t	f	f
(l)	$(m \Rightarrow n) \Rightarrow (p \Rightarrow q)$	f	f	f	f	t	t	t	t
(m)	$(m \Rightarrow (n \Rightarrow p)) \Rightarrow q$	f	f	f	f	t	t	t	t

2.2 Write truth tables to compute values for the following expressions in all states.

(a) $b \vee c \vee d$
(b) $b \wedge c \wedge d$
(c) $b \wedge (c \vee d)$
(d) $b \vee (c \wedge d)$
(e) $\neg b \Rightarrow (b \vee c)$
(f) $\neg b \equiv (b \vee c)$
(g) $(\neg b \equiv c) \vee b$
(h) $(b \equiv c) \equiv (b \Rightarrow c) \wedge (c \Rightarrow b)$

2.3 Write the duals P_D for each of the following expressions P.

(a) $b \vee c \vee true$
(b) $b \wedge c \wedge d$
(c) $b \wedge (c \vee \neg d)$
(d) $b \vee (c \wedge d)$
(e) $\neg false \Rightarrow b \vee c$
(f) $\neg b \Leftarrow b \vee c$
(g) $(\neg b \equiv true) \vee b$
(h) $(b \equiv c) \equiv (b \Rightarrow c) \wedge (c \Rightarrow b)$

2.4 For each expression $P \equiv Q$ below, write the expression $P_D \equiv Q_D$.

(a)	$p \equiv q$	(e)	$true \Rightarrow p \equiv p$
(b)	$p \wedge p \equiv p$	(f)	$false \Rightarrow p \equiv true$
(c)	$p \Rightarrow p \equiv true$	(g)	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
(d)	$p \Rightarrow q \equiv \neg p \vee q$	(h)	$p \equiv q \equiv q \equiv p$

2.5 Translate the following English statements into boolean expressions.

(a) Whether or not it's raining, I'm going swimming.
(b) If it's raining I'm not going swimming.
(c) It's raining cats and dogs.
(d) It's raining cats or dogs.
(e) If it rains cats and dogs I'll eat my hat, but I won't go swimming.
(f) If it rains cats and dogs while I am going swimming, I'll eat my hat.

2.6 Translate the following English statements into boolean expressions.

(a) None or both of p and q is $true$.
(b) Exactly one of p and q is $true$.
(c) Zero, two, or four of p, q, r, and s are $true$.
(d) One or three of p, q, r, and s are $true$.

2.7 Give names to the primitive components (e.g. $x < y$ and $x = y$) of the following English sentences and translate the sentences into boolean expressions.

(a) $x < y$ or $x = y$.
(b) Either $x < y$, $x = y$, or $x > y$.
(c) If $x > y$ and $y > z$, then $v = w$.
(d) The following are all $true$: $x < y$, $y < z$, and $v = w$.
(e) At most one of the following is $true$: $x < y$, $y < z$, and $v = w$.
(f) None of the following are $true$: $x < y$, $y < z$, and $v = w$.
(g) The following are not all $true$ at the same time: $x < y$, $y < z$, and $v = w$.
(h) When $x < y$, then $y < z$; when $x \geq y$, then $v = w$.
(i) When $x < y$, then $y < z$ means that $v = w$, but if $x \geq y$ then $y > z$ does not hold; however, if $v = w$ then $x < y$.
(j) If execution of program P is begun with $x < y$, then execution terminates with $y = 2^x$.
(k) Execution of program P begun with $x < 0$ will not terminate.

2.8 Translate the following English statement into a boolean expression. v is in $b[1..10]$ means that if v is in $b[11..20]$ then it is not in $b[11..20]$.

2.9 The Tardy Bus Problem, taken from [1], has three assumptions:

1. If Bill takes the bus, then Bill misses his appointment if the bus is late.
2. Bill shouldn't go home if Bill misses his appointment and Bill feels downcast.
3. If Bill doesn't get the job, he feels downcast and shouldn't go home.

The problem has eight conjectures:

4. If Bill takes the bus, then Bill does get the job if the bus is late.
5. Bill gets the job, if Bill misses his appointment and he should go home.
6. If the bus is late and Bill feels downcast and he goes home, then he shouldn't take the bus.