

```
1: # $Id: Makefile,v 1.1 2011-03-24 17:25:29-07 - - $
2:
3: all : foo bar
4:      echo all
5:
6: foo :
7:      echo foo
8:
9: bar :
10:     echo bar
11:
```

```
1: # $Id: Makefile,v 1.1 2011-03-24 17:25:20-07 - - $
2:
3: hello : hello.c
4:         cc hello.c -o hello
5:
6: bad : bad.c
7:         cc bad.c -o bad
8:
9: ci : Makefile hello.c bad.c
10:        cid Makefile hello.c bad.c
11:
```

```
1: # $Id: Makefile,v 1.1 2011-03-24 17:24:59-07 - - $
2:
3: all : hello
4:
5: hello : main.o hello.o
6:         cc main.o hello.o -o hello
7:
8: main.o : main.c hello.h
9:         cc -c main.c
10:
11: hello.o : hello.c hello.h
12:         cc -c hello.c
13:
14: ci : Makefile main.c hello.c hello.h
15:         cid Makefile main.c hello.c hello.h
16:
17: test : hello
18:         ./hello
19:
20: clean :
21:         - rm hello.o main.o
22:
23: spotless : clean
24:         - rm hello
25:
```

```
1: # $Id: Makefile,v 1.1 2011-03-24 17:24:36-07 - - $
2:
3: SOURCES = main.c hello.c
4: HEADERS = hello.h
5: OBJECTS = main.o hello.o
6: CHECKIN = Makefile ${SOURCES} ${HEADERS}
7: EXECBIN = hello
8:
9: all : ${EXECBIN}
10:
11: ${EXECBIN} : ${OBJECTS}
12:         cc ${OBJECTS} -o ${EXECBIN}
13:
14: main.o : main.c hello.h
15:         cc -c main.c
16:
17: hello.o : hello.c hello.h
18:         cc -c hello.c
19:
20: ci : ${CHECKIN}
21:         cid ${CHECKIN}
22:
23: test : ${EXECBIN}
24:         ./${EXECBIN}
25:
26: clean :
27:         - rm ${OBJECTS}
28:
29: spotless : clean
30:         - rm ${EXECBIN}
31:
```

```
1: # $Id: Makefile,v 1.1 2011-03-24 17:23:58-07 - - $
2:
3: MKFILE    = Makefile
4: CCOPTS    = -D__EXTENSIONS__ -Xc -v -g -xO0
5: LINTOPTS  = -D__EXTENSIONS__ -Xa -fd -m -u -x -errchk=%all,no%longptr64
6: SOURCES   = main.c hello.c
7: HEADERS   = hello.h
8: OBJECTS   = main.o hello.o
9: CHECKIN   = ${MKFILE} ${SOURCES} ${HEADERS}
10: EXECBIN   = hello
11:
12: all : ${EXECBIN}
13:
14: ${EXECBIN} : ${OBJECTS}
15:             cc ${CCOPTS} ${OBJECTS} -o ${EXECBIN}
16:
17: %.o : %.c
18:             cc ${CCOPTS} -c $<
19:
20: ci : ${CHECKIN}
21:             cid + ${CHECKIN}
22:
23: lint : ${SOURCES}
24:             lint ${LINTOPTS} ${SOURCES}
25:
26: test : ${EXECBIN}
27:             ./${EXECBIN} ; echo status = $$?
28:
29: clean :
30:             - rm ${OBJECTS}
31:
32: spotless : clean
33:             - rm ${EXECBIN}
34:
35: hello.o: hello.c
36: hello.o: hello.h
37: main.o: hello.h
38: main.o: main.c
```

```
1: # $Id: Makefile,v 1.1 2011-03-24 17:23:13-07 - - $
2:
3: MKFILE      = Makefile
4: JAVASRC     = queue.java qtest.java
5: CLASSES     = queue.class qtest.class
6: JARCLASSES  = ${CLASSES} queue\$$emptyexn.class queue\$$node.class
7: JARFILE     = qtest
8: MAINCLASS   = qtest
9: SOURCES     = ${JAVASRC} ${MKFILE}
10:
11: all : ${JARFILE}
12:
13: ${JARFILE} : ${CLASSES}
14:     echo Main-class: ${MAINCLASS} >Manifest
15:     jar cvfm ${JARFILE} Manifest ${JARCLASSES}
16:     chmod +x ${JARFILE}
17:     - rm Manifest
18:
19: %.class : %.java
20:     cid + $<
21:     javac $<
22:
23: clean :
24:     - rm ${JARCLASSES}
25:
26: spotless : clean
27:     - rm ${JARFILE}
28:
29: ci : ${SOURCES}
30:     cid + ${SOURCES}
31:
32: test : ${JARFILE}
33:     ./${JARFILE} * | cat -n
34:
```

```
1: # $Id: Makefile,v 1.1 2011-03-24 17:21:42-07 - - $
2:
3: #
4: # Article: 856 of ucsc.class.cmps112
5: # From: Daniel Keller <keller@ucsc.edu>
6: # Macros should not be replaced until they are used (not when they
7: # are first seen). E.g. The following makefile should print 'Heller':
8: #
9:
10: X      = Orwell
11: Y      = ${X}
12: X      = Heller
13: all:
14:      echo  ${Y}
15:
```

```
1: # $Id: Makefile1,v 1.1 2011-03-24 17:20:58-07 - - $
2:
3: all : foo.o
4:
5: %.o : %.c
6:     echo first %.o : %.c
7:
8: %.o : %.c
9:     echo second %.o : %.c
10:
11: foo.o: foo.c foo.h bar.h baz.h
```



```
1: # $Id: Makefile2,v 1.1 2011-03-24 17:20:58-07 - - $
2:
3: all : foo.o
4:
5: foo.o : foo.c
6:         echo specific foo.o : foo.c
7:
8: %.o : %.c
9:         echo first %.o : %.c
10:
11: %.o : %.c
12:         echo second %.o : %.c
13:
14: foo.o: foo.c foo.h bar.h baz.h
```