```
 1 #
 2 # The MIT License (MIT)
 3 #
 4 # Copyright (c) 2018 Zhou Zhi Gang
 5 # Email: keorapetse.finger@yahoo.com
 6 #
 7 # Permission is hereby granted, free of charge, to any person obtaining a copy
 8 # of this software and associated documentation files (the "Software"), to deal
 9 # in the Software without restriction, including without limitation the rights
10 # to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
11 # copies of the Software, and to permit persons to whom the Software is
12 # furnished to do so, subject to the following conditions:
13 #
14 # The above copyright notice and this permission notice shall be included in
15 # all copies or substantial portions of the Software.
16 #
17 # THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
18 # IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
19 # FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
20 # AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
21 # LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
22 # OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
23 # THE SOFTWARE.
24 #
25
26 # define any compile-time flags
27 #
28 # CFLAGS = -Wall -Wunused-variable
29 CFLAGS = -Wall
30
31 # define library paths in addition to /usr/lib
32 #   if I wanted to include libraries not in /usr/lib I'd specify
33 #   their path using -Lpath, something like:
34 #
35 LIBFLAGS = -L./lib
36
37 # define the C compiler to use
38 #
39 CC = gcc
40
41 # define any directories containing header files other than /usr/include
42 #
43 INCLUDES = -I./include
44
45 # define any libraries to link into executable:
46 #   if I want to link in libraries {libx.so or libx.a} I use the -llibname
47 #   option, something like {this will link in libmylib.so and libm.so:
48 #
49 # LIBS = -lmylib -lm
50 LIBS =
51
52 # define the C source files
53 #
54 SRCS = ${wildcard *.c}
55
56 # define the C object files
57 #
58 # This uses Suffix Replacement within a macro:
59 #   ${name:string1=string2}
60 #         For each word in 'name' replace 'string1' with 'string2'
61 # Below we are replacing the suffix .c of all words in the macro SRCS
62 # with the .o suffix
63 #
64 OBJS = ${SRCS:.c=.o}
65
66 # define the executable file
67 #
68 TARGET = twofish
69
70 #
```

```
71 # The following part of the makefile is generic; it can be used to
72 # build any executable just by changing the definitions above and by
73 # deleting dependencies appended to the file from 'make depend'
74 #
75 all: compile
76
77 compile: ${TARGET} install clean
78
79 ${TARGET}: ${SRCS}
80     ${CC} ${CFLAGS} -o $@ $^ ${LIBFLAGS} ${INCLUDES}
81
82 install:
83     mkdir -p ../bin
84     mv ${TARGET} ../bin/
85
86 clean:
87     rm -f ${OBJS}
88
```