

```
1  ### COMPILATION
2  # Le compilateur a utiliser
3  CC = g++
4  # L'editeur de lien à utiliser
5  LINK = g++
6  # Argument pour changer le nom de la sortie du compilo
7  CC_OUTPUT_FLAG = -o
8
9  # Linker la librairie tcl en fonction de l'utilisateur
10 ifeq ($(LOGNAME),element)
11     LIB_TCL = -ltcl8.4
12 else
13     LIB_TCL = -ltcl
14 endif
15
16 # Arguments pour le stade d'édit on des liens
17 LINK_FLAGS = -lstdc++ -lstdc++ -lstdc++ -lstdc++
18 # Arguments pour le stade de compilation
19 CC_COMP_FLAG = -c -std=c++11 -Wall -g -O3
20
21 # Commande d'affichage
22 PRINT = @printf
23
24 ### FICHIERS
25 # Nom du fichier de sortie
26 EXEC = bin/backlog
27 #Nom des fichiers sources
28 SRC = Mere.cpp Entree.cpp Clavier.cpp Sortie.cpp
29 # Nom du dossier sources
30 SRC_FOLDER = src/
31 # Le dossier    metre les fichiers obj s
32 OBJ_FOLDER = build/
33 # Le dossier    metre l'exec
34 EXEC_FOLDER = bin/
35
36 ### CLEAN
37 CLEAN = clean
38 # La commande pour clean
39 CLEAN_COMMAND = rm
40 # Les flags pour clean
41 CLEAN_FLAGS = -f
42 # Les fichier    clean lors d'un clean
43 ADDITIONAL_CLEAN = core $(OBJ) $(EXEC)
44
45 ### TARGET POUR LANCER LE PROGRAMME APRES LE MAKE
46 RUN = run
47
48 ### AFFICHER DU GRAS :
49 BOLD = \033[1m
50 RESET_BOLD = \033[21m
51
52 RUN_SCRIPT = ./run.sh $(EXEC)
53
54 ### VARIABLES ADDITIONNELLES CREES AUTOMATIQUEMENT
55 OBJ = $(subst .cpp,.o,$(addprefix $(OBJ_FOLDER),$(SRC)))
56
57 #####
58 #####LES CIBLES : #####
59 #####
60
61 # CIBLE ALL POUR TOUT BUILD
```

```
62 .PHONY : all
63 all : directories $(EXEC)
64
65 # CIBLE OPTIONNELLE POUR LANCER LE PROGRAMME APRES LE BUILD
66 $(RUN) : all
67 $(PRINT) "$(BOLD)EXECUTION DE $(RUN_SCRIPT)\n$(RESET_BOLD)-----\n"
68 $(RUN_SCRIPT)
69
70 # EDITION DES LIENS :
71 $(EXEC) : $(OBJ)
72 $(PRINT) "$(BOLD)EDITION DES LIENS\n$(RESET_BOLD)"
73 $(LINK) $(CC_OUTPUT_FLAG) $(EXEC) $(OBJ) $(LINK_FLAG)
74
75 # CREER LES DOSSIERS MANQUANTS
76 .PHONY : directories
77 directories : $(OBJ_FOLDER) $(EXEC_FOLDER)
78
79 $(OBJ_FOLDER) :
80 $(PRINT) "$(BOLD)CREATION DU REPERTOIRE $\n$(RESET_BOLD)"
81 mkdir -p $(OBJ_FOLDER)
82
83 $(EXEC_FOLDER) :
84 $(PRINT) "$(BOLD)CREATION DU REPERTOIRE $\n$(RESET_BOLD)"
85 mkdir -p $(EXEC_FOLDER)
86
87 # # CREER LES FICHIERS OBJETS
88 $(OBJ_FOLDER)/%.o : $(SRC_FOLDER)/%.cpp
89 $(PRINT) "$(BOLD)COMPILATION DE $\n$(RESET_BOLD)"
90 $(CC) $(CC_OUTPUT_FLAG) $@ $< $(CC_COMP_FLAG)
91
92 # PARTIE NETOYAGE
93 .PHONY : $(CLEAN)
94 $(CLEAN) :
95 $(PRINT) "$(BOLD)CLEANS\n$(RESET_BOLD)"
96 $(CLEAN_COMMAND) $(CLEAN_FLAG) $(ADDITIONAL_CLEAN)
97 $(PRINT) "\t"
98 $(CLEAN_COMMAND) -fd $(OBJ_FOLDER)
99
100 ### CIBLES DE DEPENDANCES
101 $(SRC_FOLDER)/Mere.cpp : $(SRC_FOLDER)/Config.h
102
103 $(SRC_FOLDER)/Entree.cpp : $(SRC_FOLDER)/Entree.h
104
105 $(SRC_FOLDER)/Clavier.cpp : $(SRC_FOLDER)/Clavier.h
106
107 $(SRC_FOLDER)/Sortie.cpp : $(SRC_FOLDER)/Sortie.h
108
```