

README

```
1 1. First copy the files "New_Alarm_Cond.c", "errors.h" and "makefile" files into
2   your own directory.
3
4 2. To compile the program "New_Alarm_Cond.c", use the "makefile" file provided
5   by typing 'make' into the terminal
6
7 3. Type "a.out" to run the executable code.
8
9 4. At the prompt "ALARM>", you are required to input alarm requests of either
10  Type A, B or C. A type A request being a normal alarm, Type B being a request
11  to create a thread for displaying Type As and C being a request to remove
12  type As which may result in termination of Type Bs.
13
14 Example Alarm Requests:
15
16  ALARM> 3 MessageType(2, 1) hey there!          ***   (Type A Request)
17
18  ALARM> Create_Thread: MessageType(2)           ***   (Type B Request)
19
20  ALARM> Cancel: Message(1)                      ***   (Type C Request)
21
22  (To exit from the program, type Ctrl-d.)
23
24 5. Read the report provided for a detailed explanation of how the program
25   "New_Alarm_Cond.c" works.
26
27
28 EXTRA INFORMATION:
29
30 1) A feature has been implemented to allow the user see the contents of the
31   data structures used in this program. At the prompt "Alarm>", the users can
32   input 'debug' which will activate debug mode and print out the contents of the
33   lists as well as the values for some of the semaphores. debug mode can be
34   toggled on and off by inputting the debug command. When debug mode is
35   activated, it prints out the contents of the contents of the lists and
36   semaphores every time a new alarm is processed (inserted).
37
38 2) a) For a Type A request, the first number is the display time in seconds,
39      second number is the message type, and the third is the message number.
40
41      b) For a Type B request, the only number represents the message type.
42
43      c) For a Type C request, the only number represents the message number.
44
```