a) Why are exit and cd handled as built-in commands and no child processes are forked? What would happen if you forked children for these commands?

   - Exit and cd have intentional effects on the shell “environment.” If you tried to change directory in a child, that child would cd and then die, while the main shell program would wait around oblivious to the effects of cd. In the end, cd would only change directory of the process you immediately kill thereafter, being useless. Similarly, if child processes were forked for exit commands, the children would exit and the parent would be left unaffected. Exit and cd are important to run built-in as they are meant to have a lasting effect on the program.

b) Why should you use killpg(getpid(), SIGTERM); before main() exits?

   - When you type exit to end the parent process, there may be unfinished background processes running. Because of the possibility of these unfinished processes, it is necessary to terminate the entire process group of the main process you are exiting. Otherwise, there would be no way to reap these background children and you would be left with zombies when they end.