

# Dummy Implementor's Notes

*Henry Gordon Dietz; Department of Electrical and Computer Engineering, University of Kentucky; Lexington, Kentucky*

## Abstract

*Here are three versions of the project code here that differ only in a single file and build different executables. The original is built using `bb5orig.c` and is named `bborig`. The sufficient solution is built using `bb5.c` and is named `bb`; it does as much as I expect an undergrad to have done. The graduate solution is built using `bb5grad.c` and is named `bbgrad`.*

## Make

The `Makefile` is very straightforward, and contains the usual stuff, including the ability to `make clean`, `make notes`, `make test`, and `make tar`. However, simply typing `make` will do everything important.

## Solution

I think the code speaks for itself. Then again, if I were a student submitting this, I wouldn't bet my grade on that. ;-)

## Testing

You'll see there is a test input, `test.c`. The `Makefile` automatically tries that as input to each version, placing the results in `orig`, `output`, and `grad`.

It appears that all three versions behave as they were supposed to.

## Author Biography

*Henry (Hank) Dietz earned his PhD at Polytechnic University and joined the faculty at Purdue University's School of Electrical and Computer Engineering in 1986. Since 1999, he has been a Professor and Hardyman Chair at the University of Kentucky. For some reason, he still seems to write many compilers....*