**Cpe212 2018 Quick Note Using gdb to Find Errors**

**makefile**

test: test.o

g++ -g test.o -o test

test.o: test.cpp

g++ -g -c test.cpp

**test.cpp**

#include <iostream>

using namespace std;

main()

{

int a[10];

a[1000000] = 0;

cout << "hello " << endl;

}

**example output**

kulick@eb246-38:gdb\_test> ./test

Segmentation fault (core dumped)

kulick@eb246-38:gdb\_test>

**example output run under control of gdb**

**kulick@eb246-38:gdb\_test> gdb ./test**  **binary run under gdb control**

GNU gdb (GDB) Red Hat Enterprise Linux 7.6.1-110.el7 **general information**

Copyright (C) 2013 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law. Type "show copying"

and "show warranty" for details.

This GDB was configured as "x86\_64-redhat-linux-gnu".

For bug reporting instructions, please see:

<http://www.gnu.org/software/gdb/bugs/>...

Reading symbols from /home/facstaff/kulick/wip/gdb\_test/test...done.

**(gdb) run gdb gets the program ready- run starts it**

Starting program: /home/facstaff/kulick/wip/gdb\_test/./test

Program received signal SIGBUS, Bus error. **When the error occurs gdb tells where (often)**

main () at test.cpp:8

8 a[1000000] = 0;

Missing separate debuginfos, use: debuginfo-install glibc-2.17-222.el7.x86\_64 libgcc-4.8.5-28.el7\_5.1.x86\_64 libstdc++-4.8.5-28.el7\_5.1.x86\_64

**(gdb) quit you can do more like fix errors and rerun**

**for now, we will just quit**

A debugging session is active.

Inferior 1 [process 28142] will be killed.

Quit anyway? (y or n) y

kulick@eb246-38:gdb\_test>