

**CSc 352**

C Programming  
GDB

Benjamin Dicken

**\$ man gdb**

# Making your executable compatible

Use the `-g` option when compiling with GCC

Causes the executable to include debugging information

```
$ man gcc
```

# Key commands for gdb

**break** - sets a stopping points within the code

**run** - starts the program running

**next / step** - walk through the program

**bt** - backtrace

**frame** - show information for a stack frame ( **info frame X** )

**print** - display the value of a variable / expr

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int change(int amount) {
5     if (amount == 1 || amount == 5 || amount == 10 || amount == 25) { // Base case
6         return 1;
7     }
8     if (amount > 25) {
9         return 1 + change(amount - 25); // Quarter
10    } else if (amount > 10) {
11        return 1 + change(amount - 10); // Dime
12    } else if (amount > 5) {
13        return 1 + change(amount - 5); // Nickel
14    } else {
15        return 1 + change(amount - 1); // Penny
16    }
17 }
18
19 int main(int argc, char** argv) {
20     int x = atoi(argv[1]);
21     int number_of_coins = change(x);
22     printf("Minimum coins needed: %d\n", number_of_coins);
23     return 0;
24 }
```

# Coin Program With GDB

```
lectura:> gcc -Wall -Werror -std=c11 -g coins.c -o coins
```

```
lectura:> gdb coins
```

```
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04) 9.2
```

```
Copyright (C) 2020 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-linux-gnu".
```

```
Type "show configuration" for configuration details.
```

```
For bug reporting instructions, please see:
```

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

```
Find the GDB manual and other documentation resources online at:
```

```
<http://www.gnu.org/software/gdb/documentation/>.
```

```
For help, type "help".
```

```
Type "apropos word" to search for commands related to "word"...
```

```
Reading symbols from coins...
```

```
(gdb)
```

# Coin Program With GDB

Type commands here



```
(gdb) break 5
```

```
Breakpoint 1 at 0x1178: file coins.c, line 5.
```

```
(gdb) run 27
```

```
Starting program: /home/bddicken/test/coins 27
```

```
Breakpoint 1, change (amount=27) at coins.c:5
```

```
5     if (amount == 1 || amount == 5 || amount == 10 || amount == 25) { // Base case
```

```
(gdb) bt
```

```
#0  change (amount=27) at coins.c:5
```

```
#1  0x0000555555555224 in main (argc=2, argv=0x7fffffff8f8) at coins.c:21
```

```
(gdb)
```

```
(gdb) step
8     if (amount > 25) {
(gdb) step
9     return 1 + change(amount - 25); // Quarter
```

```
(gdb) step
change (amount=21845) at coins.c:4
4     int change(int amount) {
```

```
(gdb) step
```

```
Breakpoint 1, change (amount=2) at coins.c:5
```

```
5     if (amount == 1 || amount == 5 || amount == 10 || amount == 25) { // Base case
```

```
(gdb) bt
```

```
#0  change (amount=2) at coins.c:5
```

```
#1  0x0000555555551aa in change (amount=27) at coins.c:9
```

```
#2  0x000055555555224 in main (argc=2, argv=0x7fffffff8f8) at coins.c:21
```

```
(gdb)
```



## (gdb) info frame 0

Stack frame at 0x7fffffff7c0:

rip = 0x55555555178 in change (coins.c:5); saved rip = 0x555555551aa

called by frame at 0x7fffffff7e0

source language c.

Arglist at 0x7fffffff798, args: amount=2

Locals at 0x7fffffff798, Previous frame's sp is 0x7fffffff7c0

Saved registers:

rbp at 0x7fffffff7b0, rip at 0x7fffffff7b8

## (gdb) info frame 1

Stack frame at 0x7fffffff7e0:

rip = 0x555555551aa in change (coins.c:9); saved rip = 0x55555555224

called by frame at 0x7fffffff810, caller of frame at 0x7fffffff7c0

source language c.

Arglist at 0x7fffffff7b8, args: amount=27

Locals at 0x7fffffff7b8, Previous frame's sp is 0x7fffffff7e0

Saved registers:

rbp at 0x7fffffff7d0, rip at 0x7fffffff7d8

## (gdb) info frame 2

Stack frame at 0x7fffffff810:

rip = 0x55555555224 in main (coins.c:21); saved rip = 0x7ffff7de20b3

caller of frame at 0x7fffffff7e0

source language c.

Arglist at 0x7fffffff7d8, args: argc=2, argv=0x7fffffff8f8

Locals at 0x7fffffff7d8, Previous frame's sp is 0x7fffffff810

Saved registers:

rbp at 0x7fffffff800, rip at 0x7fffffff808

(gdb)

# Debug

- Download `test.c` and `makefile` from the class website
- Without modifying the makefile or C file, determine:
  - What could cause this program to crash?
  - Why?
  - Use GDB
- I'll give you 5-7 minutes to download, test, explore with GDB, then we can discuss