CSE 306 Software Quality in Practice

Dr. Carl Alphonce
alphonce@buffalo.edu
343 Davis Hall
EXERCISE
(from last time)
Switch to b, edit F2 to create conflict (edit same part of F2 as you did on branch a, but change it in a different way), merge b into main (switch to main, then `git merge b`).
STARTING POINT
git checkout b
workspace

index

staging

README.md

F1

F2''

F3

F4

t

c

c

c

c

c

c

c

c

c

edit F2 to create conflict with F2'
git add F2''
git commit -m "..."
git checkout main
Blob created with conflict markings.

`git merge b`
edit F''' to resolve conflict, creating F2^4
add to staging
Merge is completed (this time a non-fast forward)

git commit -m "..."
Question

- Is there a difference between
  - merging a into b
  - merging b into a
Question

- Is there a difference between
  - merging a into b
  - merging b into a

YES!
merging b into main
main moves, b does not
merging main into b
b moves, main does not
also:
parent order differs
The parents of a commit are ordered.

A =      = A^0
B = A^   = A^1     = A~1
C = A^2  = A^2
D = A^^  = A^1^1   = A~2
E = B^2  = A^^2
F = B^3  = A^^3
G = A^^^ = A^1^1^1 = A~3
H = D^2  = B^^2    = A^^^2  = A~2^2
I = F^   = B^3^    = A^^3^
J = F^2  = B^3^2   = A^^3^2
merging b into main
HEAD^1 → C, HEAD^2 → B
merging main into b
HEAD^1 → B, HEAD^2 → C
• Document baseline approach to SW development in a team environment

• What are we looking for?
  Documentation of process.

• Some teams did not collaborate/communicate well.
  Something to work on: how can you (as an individual & as a team) encourage/ensure collaboration and communication?
Learning outcomes of course

(I) Employ static and dynamic analysis tools to detect faults in a given piece of software.

(II) Employ profiling tools to identify performance issues (both time and memory) in a given piece of software.

(III) Employ testing frameworks to write tests that fail in the presence of software faults, and pass otherwise.

(IV) Employ a structured, methodical approach to detecting, testing, identifying and correcting software faults.

(V) Work productively as a member of a software development team.
Think broadly
Think broadly
build to LPR
Think broadly
build to LPR
apply in other courses
Think broadly
build to LPR
apply in other courses
showcase to potential employers
EXP01

- Released this coming Monday
- Team-based: same teams as for PRE
- Clone repo via GitHub at usual so course staff can view
- Learning goals:
  - show you can apply process
  - show you can use tools effectively
  - show you can engage in teamwork
    - communication and collaboration are key
  - More to come between EXP01 and EXP02
More gdb commands

- C-x C-a toggle between a "graphical" and line-based UI
- break <line> (e.g. break 31)
- info b (list breakpoints)
- c (continue to next breakpoint), c 10
- watch <variable> (e.g. watch i)
  - https://sourceware.org/gdb/current/onlinedocs/gdb/Set-Watchpoints.html#Set-Watchpoints

Looking at source code:
- list line#
- list function
- disassemble /m

Looking at data:
- print
- examine (x)
- x /s name, x/48c name (addresses in hex!)
- https://sourceware.org/gdb/current/onlinedocs/gdb/Memory.html#Memory
```c
#include <stdlib.h>
#include <string.h>
#include <stdio.h>

int main(int argc, char * argv[]) {
    if (argc != 2) {
        printf("Please give one numeric argument.\n");
        return 1;
    }

    int limit = atoi(argv[1]);

    char * string,* name;
    name = malloc(3 * sizeof(*name));
    string = malloc(9 * sizeof(*string));
    name[0] = '@';
    name[1] = '$';
    name[2] = '\0';
    string[0] = 's';
    string[1] = 'e';
    string[2] = 'r';
    string[3] = 'e';
    string[4] = 'n';
    string[5] = 'i';
    string[6] = 't';
    string[7] = 'y';
    string[8] = '\0';
    printf("string has length %zu and is %s.\n",strlen(string),string);
    printf("name has length %zu and is %s.\n",strlen(name),name);
    for (int i=3; i<limit; i++) {
        name[i] = (char) ('a'+((i-3)%26));
    }
    name[limit] = '\0';
    printf("string has length %zu and is %s.\n",strlen(string),string);
    printf("name has length %zu and is %s.\n",strlen(name),name);
    return 0;
}
```

GitHub Classroom link for this code: [https://classroom.github.com/a/MFvjkbsJ](https://classroom.github.com/a/MFvjkbsJ)
(also posted on schedule page of course website)