final project

Erfan Tajik

January 29, 2024

Contents

1 Git and Github

1.1 Repository Initialization and Commits

i make new repo in github, then i go to actions section and add new action and copy action file from milli repo and paste it, then i clone the project on my system, create new doc.tex file and push it

1.2 GitHub Actions for LaTeX Compilation

To create a lightweight tag (just a pointer to a commit), use:
$git tag [tagname]
To push a single tag to GitHub, use:
$git push origin [tagname]
by the action that we add in last section when we push in github with a new tag, github automaticaly compile our file and release it.

2 Exploration Tasks

2.1 Vim Advanced Features

1. Macros - Vim allows you to record and replay sequences of commands as macros. This allows you to automate repetitive tasks. To record a macro, press q followed by a register (a-z), perform your commands, then press q again to stop recording. To replay the macro, press @ followed by the register.

2. Vimscript - Vim has its own scripting language called Vimscript that allows you to customize and extend the editor. You can write Vimscripts to create custom commands, mappings, autocommands, and more. Vimscript files have a .vim extension and are loaded automatically on startup.

3. Splits and Tabs - Vim allows you to view and edit multiple files or views of the same file in a single session. You can split the window horizontally or vertically to show different split panes using :split or :vsplit. Within each split you can open a file. Vim also supports tabs, allowing you to open files or splits in separate tab pages accessed through :tabnew or :tabedit. You can navigate between splits and tabs to easily work across multiple files/views.

2.2 Memory profiling

2.2.1 Memory Leak

A memory leak occurs when memory is allocated dynamically but is not freed when it is no longer needed. This often happens when pointers to allocated memory are lost or overwritten before the memory is freed. Common causes in C include:

- Forgetting to free memory that was previously allocated with malloc()/calloc() /realloc(). Any memory not explicitly freed will remain allocated until the program exits.
- Failing to free memory before overwriting a pointer that points to it. If the only pointer to a block of memory gets overwritten, that memory can no longer be accessed to free it.
- Continuously allocating new memory without freeing old memory that is no longer needed. Over time this can cause the program to run out of available memory.
- Freeing the same memory block twice. This can corrupt the heap memory management structures, making future allocations fail.