Cw-final 403521096

Bahman
Sadra-bakhshi

1 GIT AND GITHUB

1.1 Creating and Using a Repository

First, choose the repository section of the GitHub website. Then select a suitable name for the repository, e.g., `cw-final`. Next, clone and set up the GitHub repository with the following commands:

```
echo "#␣cw-final403521096" >> README.md
git init
git add README.md
git commit -m "first␣commit"
git branch -M main
git remote add origin https://github.com/sadra243/cw-final403521096.git
git push -u origin main
```

1.2 Setting Up GitHub Actions Workflow

In the repository, create a directory `.github/workflows`. Inside `.github/workflows`, create a YAML file, e.g., `compile-latex.yml`, and use the following workflow:

```
name: Compile LaTeX Document

on:
  push:
    branches:
      - main  # Specify branches to trigger the workflow
  pull_request

jobs:
  build:
    runs-on: ubuntu-latest
    steps:
    - name: Checkout repository
      uses: actions/checkout@v3
    - name: Set up TeX Live
      uses: dante-ev/latex-action@v2
    - name: Compile LaTeX document
      run: |
        latexmk -pdf -interaction=nonstopmode -output-directory=build main.
    - name: Upload compiled PDF
      uses: actions/upload-artifact@v3
      with:
        name: compiled-pdf
        path: build/main.pdf
```

Workflow Explanation: The workflow triggers on `push` and `pull_request` events for the main branch. It uses the following actions:

- `actions/checkout`: Checks out the repository code.
- `dante-ev/latex-action`: Sets up the TeX Live environment for LaTeX dependencies.
- `actions/upload-artifact`: Uploads the compiled PDF as a downloadable artifact.

2 Vim Commands

2.1 Split, Tabs, and Buffers

- Splitting: Use `vsplit` or `split`.
- Tabs: Open new tabs with `:newtab`, switch with `tabnext`.
- Buffers:
  - `:ls` - Lists buffers.
  - `b<number>` - Switch to a buffer.

2.2 Macros

- Start recording commands with `q`.
- Stop recording with `@`.

3 Memory Management

3.1 Memory Leaks

A memory leak occurs when dynamically allocated memory (e.g., using `malloc`, `calloc`, or `realloc` in C) is not released using `free`. For example:

```c
int *ptr = (int *)malloc(sizeof(int));
*ptr = x;  // Memory allocated but never freed.
```

3.2 Valgrind

Valgrind is a tool for memory debugging, profiling, and leak detection in C/C++ programs.

Purpose:

- Detect memory leaks.
- Track invalid memory access.
- Provide detailed reports of issues.

4 Fuzzy Search

4.1 Overview

Fuzzy search matches approximate patterns in data.

4.2 Commands

- `fd --extension pdf` - List PDF files.
- `fd --extension pdf | fzf` - Filter results using `fzf`.
- Open PDF with `zathura`:

```
zathura /path/to/selected/file.pdf
zathura $(find /path/to/search –name ”desired_file.pdf”)
```

5 Using GitHub

5.1 Sections 3.1 and 3.2

Follow GitHub setup and guidelines for project management.

5.2 Development Opportunities

GitHub offers numerous projects to enhance skills.