Computer Science Mentor Scheme

Linux/Git Activity

If you haven't done Welcome to the Machine, please work on this instead. It will be more useful to you.

In your Linux environment (for most of you, the **Ubuntu VM**) you installed, work through the tasks below. You may not get them all done, but get as far as you can.

Hints are available on the other side of this sheet.

School Gitlab Server:

https://projects.cs.nott.ac.uk

1. Configure your VirtualBox environment (if using one) by increasing the RAM, video memory and CPU core count. We recommend **4GB of RAM** or more, **128MB** of video memory, and **2+** cores.
2. Make sure you have the following packages installed:
   - **build-essential** (required for PGA)
   - If using virtualbox:
     - **virtualbox-guest-dkms**
     - **virtualbox-guest-utils**
     - **virtualbox-guest-x11**
   - **vscode** (optional)
3. Create a **personal access token** on the school GitLab server.
4. On your Linux machine, **create a folder** somewhere. It's not important, although '*~/Code/MentorActivity*' would be a good location.
5. **Initialise an empty repository** in this folder.
6. **Add a README.md** file, include some text, then **commit this file** to the repository with the message "Initial commit".
7. On the school GitLab, **create a new empty project**. Add this project as a remote to your new local repository, then **push your commit to the remote branch**.
8. In the local repository, **create a new branch** called "add-code".
9. **Create a file called "helloworld.c"**. Add the following code, then **commit it to the branch**:

```c
#include <stdio.h>

int main(int argc, char **argv) {
    printf("Hello, world!\n");
    return 0;
}
```

10. **Merge this branch** back into your master branch, then **push the changes to the remote repository**.
11. **Compile and run** your code with:

```
gcc helloworld.c --ansi --pedantic-errors -Wall -o helloworld
```

12. Run your program with `./helloworld`, and **show a mentor**.

Hints

1. You can access the configuration menu for your VM by selecting it in the list and clicking the orange gear.
2. - `sudo apt update` to update the package database.
   - Use `sudo apt install [packages...]` to install packages.
   - To install Visual Studio Code, use `snap install --classic vscode`.
3. You can access the menu to do this by going to the home page of the GitLab site, clicking your avatar in the top right and selecting “Edit Profile”. You should see “Access Tokens” on the left navigation bar. Use this as your password when prompted by git.
4. You can make folders using the `mkdir` command, and move into them using `cd`.
5. You can initialise an empty repository using `git init`.
6. - Make sure to run `git add README.md` before committing.
   - Use `git status` to make sure everything’s right.
   - Make the commit by running `git commit -m "<commit-message-here>"`
7. On the home page, you can create a new project by clicking in the top right.
8. To create a new branch, use `git branch add-code`. You can then move to that branch using `git checkout add-code`.
9. You can do this with any text editor. If you’re struggling, run `gedit helloworld.c`
10. You can merge the branch by first moving to the master branch with `git checkout`, then using `git merge add-code`.
11. N/A
12. N/A