FREE SOFTWARE FOUNDATION

<u>The Problem</u>: Though freedom-respecting (free/libre) programs predate their non-free counterparts, non-free programs have become commonplace, especially in educational environments. Although adhering to the free software philosophy would benefit all, academia should already be focused on using exclusively libre programs. Thus, for this project, I will create an opinion-editorial piece, published physically in *The Huntington News*, explicating the free software movement through the scope of education, in an attempt to convince the reader to switch to more free alternatives, and possibly even begin to support the movement. This is aimed at readers involved in academia, whether student, professor, or researcher.

Author's Note

Beginning the revision process, I found it best to begin by simply re-reading the entire passage. As noted by a peer, some areas appeared to be wordy and a bit too technical for someone being introduced to free software. As such, I decided it would be best to break up the article into more easily readable sections, and then focus on clarifying the ideas present within these sections. Some terms were pointed out as too technical or field specific (such as chief GNUissance), and, as such, I adjusted the piece to avoid using such terms. Most importantly, a peer mentioned that the word "clueless" generally has a negative connotation. Though I did not read it this way when I initially wrote the piece, re-reading the passage did seem to indicate that many people may be alienated by this kind of terminology. Thus, I changed "clueless" to "uninformed." Finally, my mention of Stallman's initials, RMS, seemed to be unnecessary initially, as I continued to refer to him as Stallman, but the new revision utilizes RMS instead.

It was also pointed out that the document seems to have very limited white space, and that more spacing should be used. Though, in general, I agree that adequate use of white space is necessary, I made the decision that, since I am trying to mimic a printed article (such as a newspaper), which generally also utilize a very text-dense format, I would keep this arrangement.

As suggested, I decided to expand the problem statement a bit as well. Although the changes were minor, I thought it would be logical to add explicit statements related to where I envision this work would be published, and with what target audience. Finally, I thought it necessary to include minor edits related to overall phrasing and complexity of sentences, in order to make the piece more accessible to less technically-oriented readers.

Introduction

WAIT! . . . see, that is exactly what your computer *should* be doing . . . a computer waits to receive instructions — if it is not computing, it is waiting. But who is supplying these instructions? Is it you, or is it someone else? Your digital freedom may be at stake! With a rise in the prevalence and complexity of technology, come questions on the ethics. These questions range from privacy and tracking protection to whether companies should be able to obfuscate their code. The underlying argument of the Free Software Foundation, abbreviated FSF, essentially lies in a dichotomy: if the user does not control their system, then the system controls them. Nowadays, companies such as Microsoft, Google, Apple, and many more profit off of uninformed users by forcing them to pay, perpetually, not for the product, but for a license to be able to use said product. This immoral business model limits the users' ability to properly use the software, thereby dividing the user community and establishing the dominance of the company over the user.

The Inception of Free Software

Richard Stallman (generally referred to by his initials, RMS), the founder of the FSF, has been fighting for the rights of users since the early 1970s, when he graduated with a degree in physics from Harvard University, and joined the MIT Artificial Intelligence Laboratory. RMS argues that, for a piece of software to be free (that is, free as in freedom, not price), it must follow the four freedoms: first (or the zeroth term, the way RMS numbers it), a user must be able to run the program for any purpose. Second, a user must be able to study the program and change it to their needs. Third, a user must be able to redistribute the program to help out their neighbor. Finally, fourth, a user must be able to improve the program, and release said improvements to other members of the community.

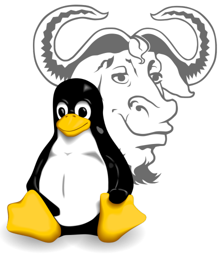
To provide an analogy, imagine owning a physical book. Your ability to run the code as you wish would be akin to your ability to read the book whenever you wish. The second freedom would be similar to being able to read the book, and write in the margins (whether it be notes, comments, possible changes, etc.). The third freedom would be like lending this book to your neighbor. Finally, the fourth freedom would be similar to a combination of freedoms two and three: rewriting a new revision of the book, and distributing it to other people you know.

As such, now with codified requirements, RMS set out to create a license to benefit the community. Codified requirements were established as the GPL, or General Public License, which has since reached the third version. This is the foremost free software license (some non-FSF licenses exist), and is endorsed by the FSF.

Free as in Freedom

Dr. Richard M. Stallman

Of course, many questions come with the idea of "free software." For example, it is frequently asked whether an/a individual/company may receive compensation (usually in the financial form) for creating the piece of software. Of course, this is not only allowed, but also encouraged! For example, a widely-used, real-life example lies in the Canvas Learning Management System, utilized by Northeastern University. The Canvas code itself is licensed under the AGPL (Affero GPL) license, which is commonly used for web-based software. The question about money is often asked because the main point of the FSF is misconstrued. The FSF refers to free software as relating to liberty, not price. Furthermore, the Free Software Movement concerns us, as students, in that non-free software has become ubiquitous in today's classrooms. One major example is Zoom, which has faced hundreds of privacy and rights infringement lawsuits. As is well known, due to the rise of distance learning, many schools have been faced with a transition to online learning. With this move came the need for some kind of replacement for in-person learning, and, as a result, learning establishments (as well as many companies) turned to Zoom. A great, decentralized, GNU variant is Jitsi Meet. Unlike Zoom, the user is able to host the server for meetings on their machine, without the need for an account. This means that Jitsi is private and anonymous, which makes it more efficient, as well as a safer tool for students and teachers to use. In RMS's article, entitled "*Why Schools Should Exclusively Use Free Software*," RMS outlines the main reasons for schools to transfer to free systems, as well as the fiscal freedom as a byproduct of using GNU and/or free software.


GNU/Linux Symbols

RMS argues that schools have a social responsibility, "to teach students to be citizens of a strong, capable, independent, cooperating and free society." As such, by using non-free software, schools are forming a dependency on the aforementioned software, thereby creating unprepared, dependent students.

Misconceptions

Often, students bring up the idea of an "educational license" that many companies, such as Microsoft, Mathworks (through MatLab), and Google give out "for free" to students. Although it may appear that this is free in price, it is ultimately the students who pay for it. These licenses are purposely made so that people are hooked to using their products, which, in the long run, will make these companies money, as students graduate and begin to pay for their licenses. Therefore, it is in the best interests of society worldwide to teach students using free software and programs.

Interested in learning more? Check out the links below:

Free Software Foundation Official Website

GNU Official Website

Richard Stallman's Official Website

Written and Edited by Michael Brodskiy

We can't live, work, or learn in freedom unless the software we use is free.