# FSE 201: Engineering UGTA
# UGTA Self-Selected Contribution Template

**UGTA Name: Jacob Hreshchyshyn**
**UGTA Course: CSE240**
**Course Professor: Dr. Javier Gonzalez-Sanchez**
**Session: C**

---

**Activity Title:**
**Expected Points: 1**

---

**Activity Summary:**

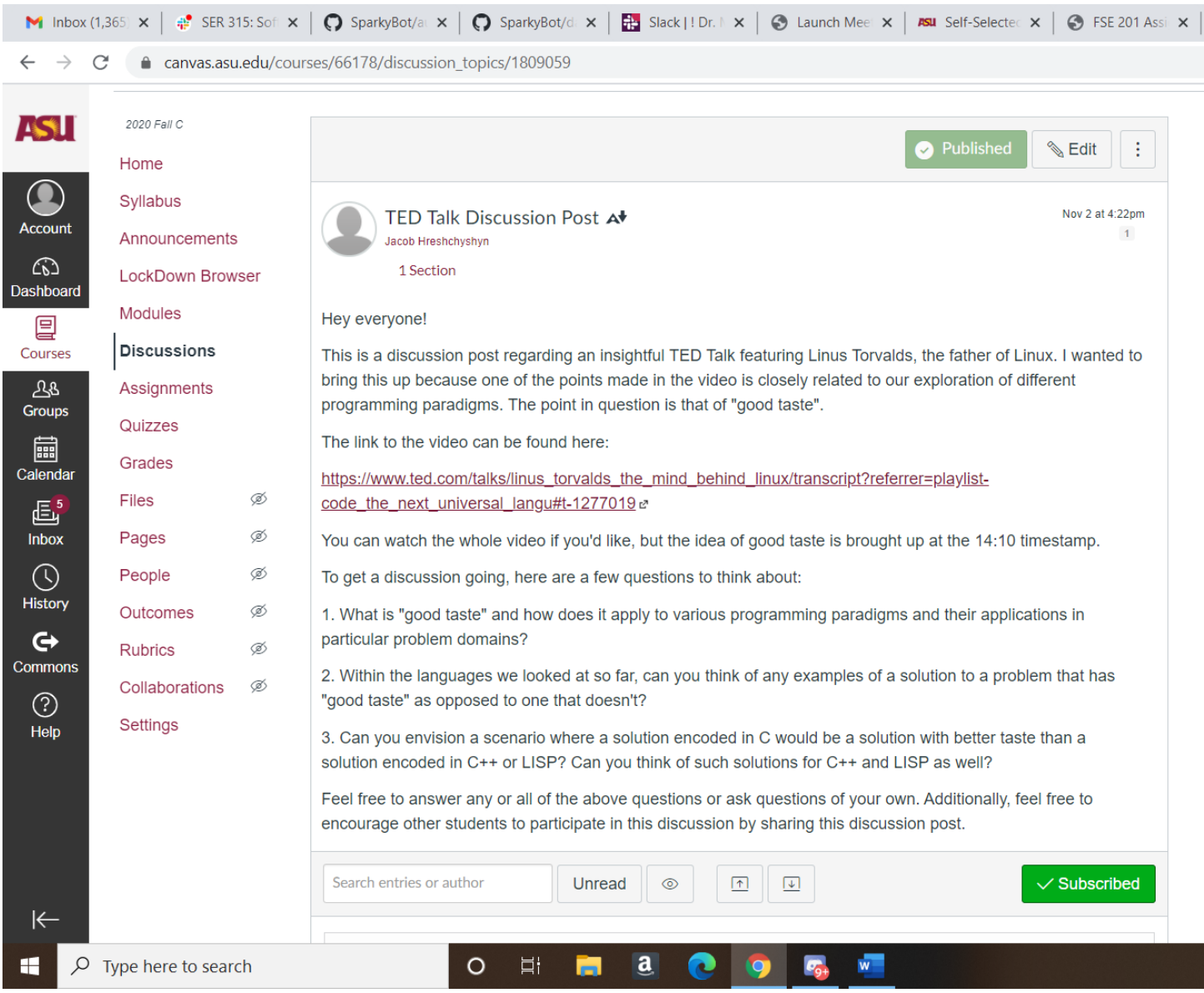
At the time of this submission, only one student has responded to the discussion post. However, the student who did respond seemed pleased with the TED Talk recommendation. He was grateful for learning about Linus Torvald's involvement in the development of Git and demonstrated a desire to engage with the questions I posed in the discussion post. Next time, more incentive might be required in order to get more students to participate in the discussion. However, it was encouraging to see student participation occur organically.

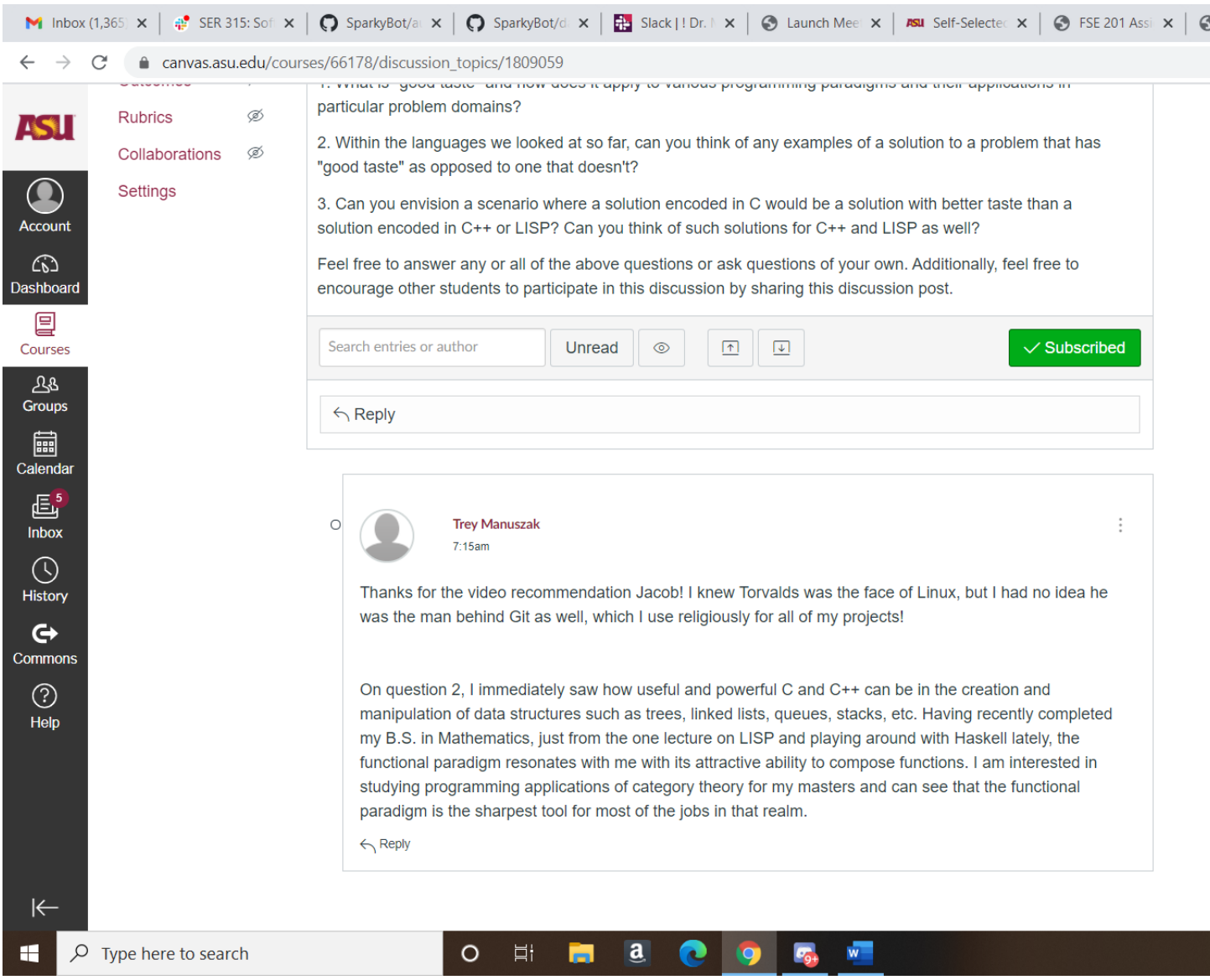
**Activity Supporting Materials and Documentation:**

Link to discussion post:
https://canvas.asu.edu/courses/66178/discussion_topics/1809059

Link to TED Talk video:
https://www.ted.com/talks/linus_torvalds_the_mind_behind_linux/transcript?referrer=playlist-code_the_next_universal_langu#t-1277019

Discussion post pictures:


Inbox (1,365
SER 315: Soft
SparkyBot/a
SparkyBot/d
Slack | ! Dr.
Launch Mee
Self-Selecte
FSE 201 Assi
canvas.asu.edu/courses/66178/discussion_topics/1809059
2020 Fall C
Home
Syllabus
Announcements
LockDown Browser
Modules
Discussions
Assignments
Quizzes
Grades
Files
Pages
People
Outcomes
Rubrics
Collaborations
Settings
Account
Dashboard
Courses
Groups
Calendar
Inbox
History
Commons
Help
Published
Edit
TED Talk Discussion Post
Jacob Hreshchyshyn
1 Section
Nov 2 at 4:22pm
Hey everyone!
This is a discussion post regarding an insightful TED Talk featuring Linus Torvalds, the father of Linux. I wanted to bring this up because one of the points made in the video is closely related to our exploration of different programming paradigms. The point in question is that of "good taste".
The link to the video can be found here:
https://www.ted.com/talks/linus_torvalds_the_mind_behind_linux/transcript?referrer=playlist-code_the_next_universal_langu#t-1277019
You can watch the whole video if you'd like, but the idea of good taste is brought up at the 14:10 timestamp.
To get a discussion going, here are a few questions to think about:
1. What is "good taste" and how does it apply to various programming paradigms and their applications in particular problem domains?
2. Within the languages we looked at so far, can you think of any examples of a solution to a problem that has "good taste" as opposed to one that doesn't?
3. Can you envision a scenario where a solution encoded in C would be a solution with better taste than a solution encoded in C++ or LISP? Can you think of such solutions for C++ and LISP as well?
Feel free to answer any or all of the above questions or ask questions of your own. Additionally, feel free to encourage other students to participate in this discussion by sharing this discussion post.
Search entries or author
Unread
Subscribed
Type here to search


Inbox (1,365)
SER 315: Soft
SparkyBot/a
SparkyBot/d
Slack | ! Dr.
Launch Mee
Self-Selecte
FSE 201 Assi
canvas.asu.edu/courses/66178/discussion_topics/1809059
Rubrics
Collaborations
Settings
Account
Dashboard
Courses
Groups
Calendar
Inbox
History
Commons
Help
particular problem domains?
2. Within the languages we looked at so far, can you think of any examples of a solution to a problem that has "good taste" as opposed to one that doesn't?
3. Can you envision a scenario where a solution encoded in C would be a solution with better taste than a solution encoded in C++ or LISP? Can you think of such solutions for C++ and LISP as well?
Feel free to answer any or all of the above questions or ask questions of your own. Additionally, feel free to encourage other students to participate in this discussion by sharing this discussion post.
Search entries or author
Unread
Subscribed
Reply
Trey Manuszak
7:15am
Thanks for the video recommendation Jacob! I knew Torvalds was the face of Linux, but I had no idea he was the man behind Git as well, which I use religiously for all of my projects!
On question 2, I immediately saw how useful and powerful C and C++ can be in the creation and manipulation of data structures such as trees, linked lists, queues, stacks, etc. Having recently completed my B.S. in Mathematics, just from the one lecture on LISP and playing around with Haskell lately, the functional paradigm resonates with me with its attractive ability to compose functions. I am interested in studying programming applications of category theory for my masters and can see that the functional paradigm is the sharpest tool for most of the jobs in that realm.
Reply
Type here to search