

d-khan / assembly

Q

Type [🔗](#) to search

>_

+ ▾

🕒

🔗

📧

<> Code

🕒 Issues

🔗 Pull requests

🎬 Actions

📁 Projects

🛡 Security

📈 Insights

📁 Files

🔗 main ▾

+ 🔍

🔍 Go to file

t

> 📁 Boolean_algebra

> 📁 Introduction to AL, variables an...

> 📁 arithmetic-instructions

> 📁 computer_organization

✓ 📁 condition-instructions

📄 Activity.html

📄 Activity.md

📄 Lecture.html

📄 Lecture.md

> 📁 file-management

> 📁 functions

> 📁 hello-world

> 📁 logical-instructions

> 📁 loops-arrays

> 📁 memory_organization

> 📁 numberingsystems

> 📁 procedures

📄 .DS_Store

📄 All Assembly Code

[Documentation](#) • [Share feedback](#)

assembly / condition-instructions / Activity.md 📄

🧑‍💻 d-khan

Update Activity.md

bfe4b86 · 6 months ago

🕒 History

assembly / condition-instructions / Activity.md

↑ Top

Preview

Code

Blame

59 lines (37 loc) · 1.78 KB

🧑‍💻 Code 55% faster with GitHub Copilot

Raw

📄

⬇

✎ ▾

☰

Activity - Conditional instructions [🔗](#)

Objective [🔗](#)

Learn to perform conditional instructions in Assembly Language.

Prerequisites [🔗](#)

- Before doing the lab, please review the lecture.
- This lab will only work if you run the code on a Linux platform using an Intel x86 platform. The online assembler will not work due to the lack of debugging features.
- Knowledge of how to run assembly code using nasm assembler in Linux OS.
- Knowledge of how to debug an assembly code using `gdb` .

Task [🔗](#)

Perform the following tasks:

1. Generate a sequence of even or odd numbers up to 20.
2. Find the largest number among the **five integer numbers**. Use initialize variables, and the output goes to the largest un-initialize variable.

Debugging parameters [🔗](#)

I recommend using the following debugging parameters to display results. See the lecture contents for the explanation.

```
layout asm
layout regs
watch (int) largest
break _start
run
stepi
```

What to submit? [🔗](#)

1. Draw a flowchart of your thought process. I found this [online flowchart website](#) very useful. However, you can use any application of your choice. (1 mark)
2. What were your challenges in performing the lab (from design to the implementation phases)? (1 mark)
3. Working and error-free code. (8 marks)

How to submit it? [🔗](#)

- Upload the work in Canvas and clearly define your responses.
- Upload the code in **.txt** format and include comments to describe the code.
- **Do not compress or zip your work.**

Deadline [🔗](#)

The deadlines are posted on the Syllabus as well as on Canvas.

Rubric [🔗](#)

- All the questions are answered, and the working code is submitted. (Grade 100%)
- Questions are partially answered, and the code has errors or incorrect output. (Grade 50%)

Last updated: Mar 2023 by Dr. Danish Khan.