F28HS with RBL

Responsive Blended Learning in 2021

Block teaching

- Weeks 1-3
  - C programming
  - Dr Rob Stewart
- Weeks 4-5
  - Assembly programming
  - You've seen ARM assembly in Introduction to Computer Systems (F27CS)
  - Dr Rob Stewart
- Weeks 6-11
  - Systems programming (combining C and Assembly)
  - Dr Hans-Wolfgang Loidl
- Week 12: Revision week

Assessment weightings

- 30% C programming
  - 25% Coursework 1
  - 5% class test
- 30% Assembly programming
  - Coursework 2
- 40% Systems programming
  - Coursework 3
  - Involves both C and Assembly programming

Assessment dates

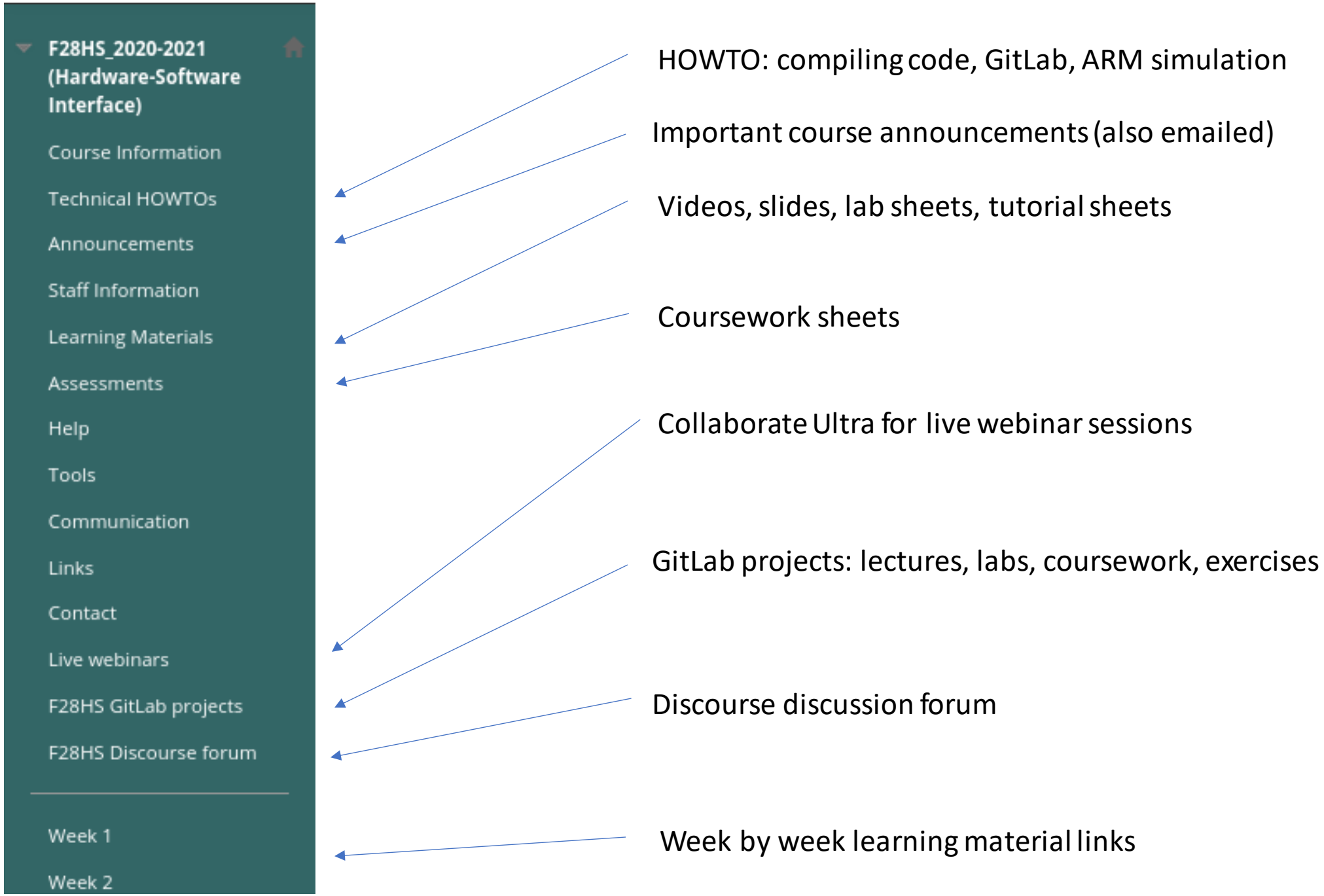
Assessment	Date
C class test	Week 4
Coursework 1 (C)	Week 5, Thursday 11th February
Coursework 2 (Assembly)	Week 7, Thursday 25th February
Coursework 3 (Systems)	Week 11, Thursday 25th March

Effort hours

10 hours per week for a 15 credit course

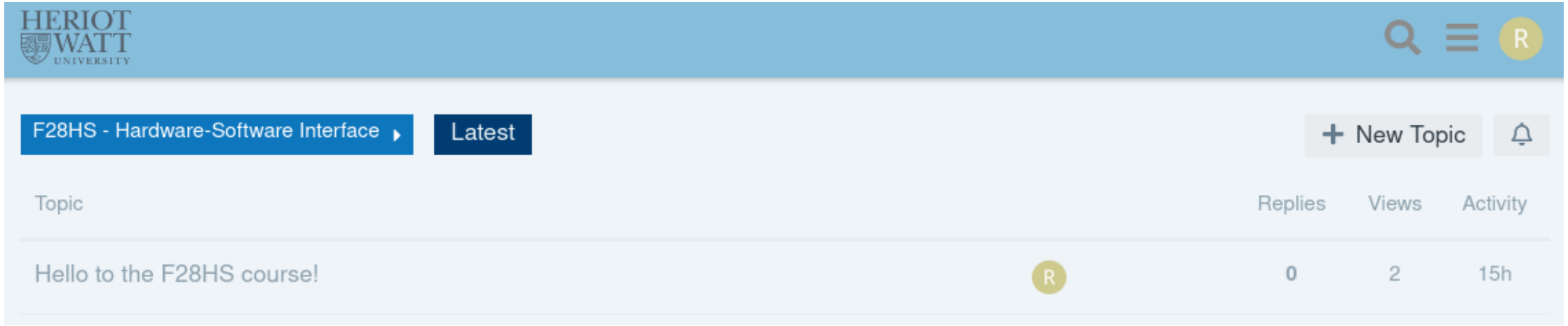
Activity	Hours	Mode	Help
Watch pre-recorded lectures	2	In your own time	During Monday/Tuesday live sessions
Live sessions	2	Monday 11am, Tuesday 12pm	During Monday/Tuesday live sessions
Lab sheets	2	In your own time	GitLab "@all help me", Discourse
Coursework	3	In your own time	GitLab "@all help me", Discourse
Additional reading	1	In your own time	Discourse
Total =	**10**		

This course requires a lot of programming practise in your own time.


F28HS_2020-2021 (Hardware-Software Interface)
Course Information
Technical HOWTOs
Announcements
Staff Information
Learning Materials
Assessments
Help
Tools
Communication
Links
Contact
Live webinars
F28HS GitLab projects
F28HS Discourse forum
Week 1
Week 2
HOWTO: compiling code, GitLab, ARM simulation
Important course announcements (also emailed)
Videos, slides, lab sheets, tutorial sheets
Coursework sheets
Collaborate Ultra for live webinar sessions
GitLab projects: lectures, labs, coursework, exercises
Discourse discussion forum
Week by week learning material links

Discourse

Group discussion **strongly encouraged** in this course



Questions about

- Pre-recorded lectures
- Lab sheets
- Tutorials
- Coursework
- GitLab

Questions about

- C programming
- Assembly programming
- Systems programming

https://discourse.macs.hw.ac.uk

Register here: https://discourse.macs.hw.ac.uk/g/Y2-F28HS

Communication

- General technical questions: Discourse
- Logistical questions: Discourse
- Specific programming coursework questions
  - Discourse
  - GitLab: "@all help me..."
  - Interaction with teaching team and lab helpers
- Office hours
  - Dr Rob Stewart: 2-3pm Wednesdays. MS Teams by appointment.
  - Dr Hans-Wolfgang Loidl: 2-3pm Thursdays. MS Teams.

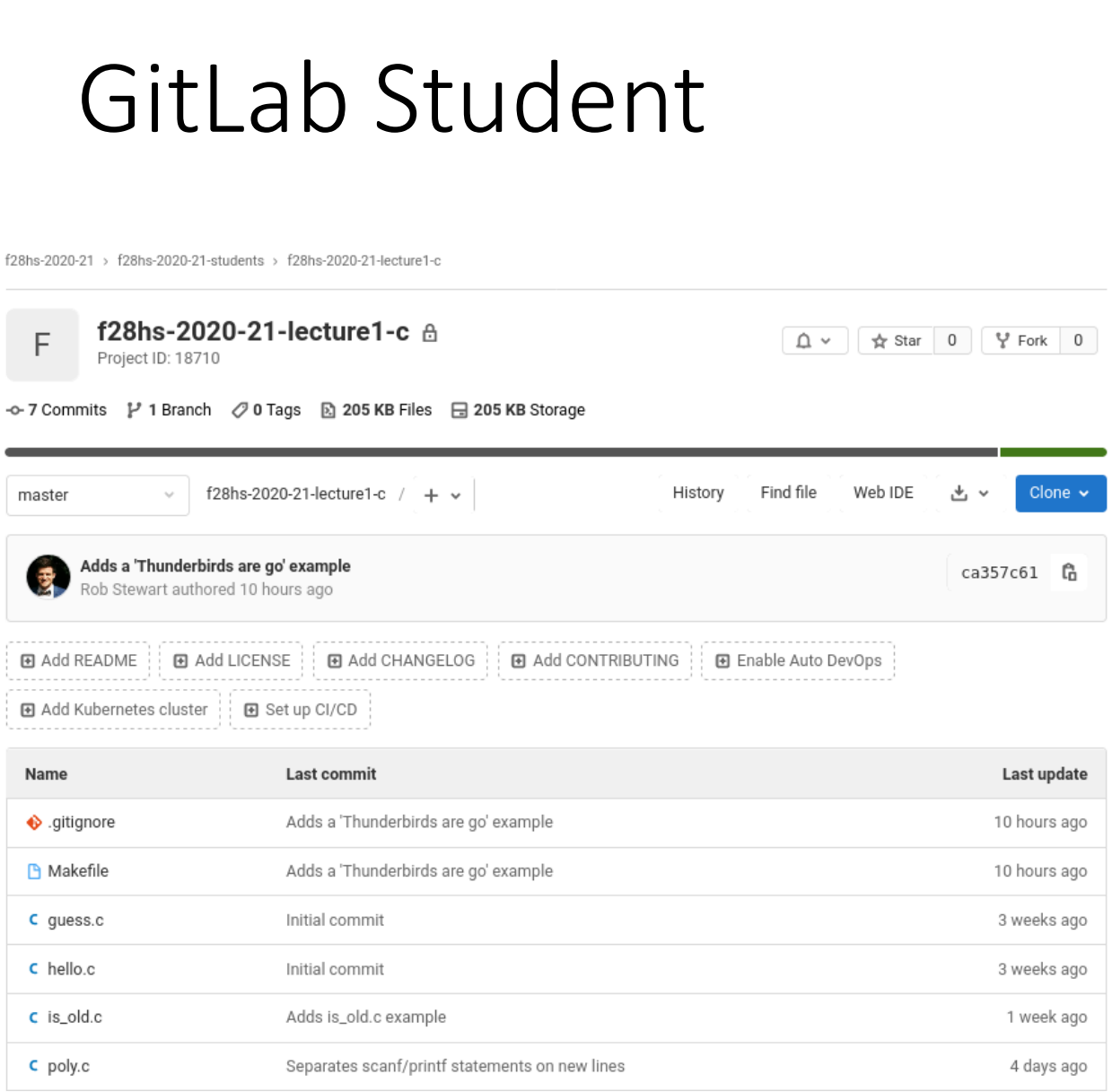
Email is not used in this course.

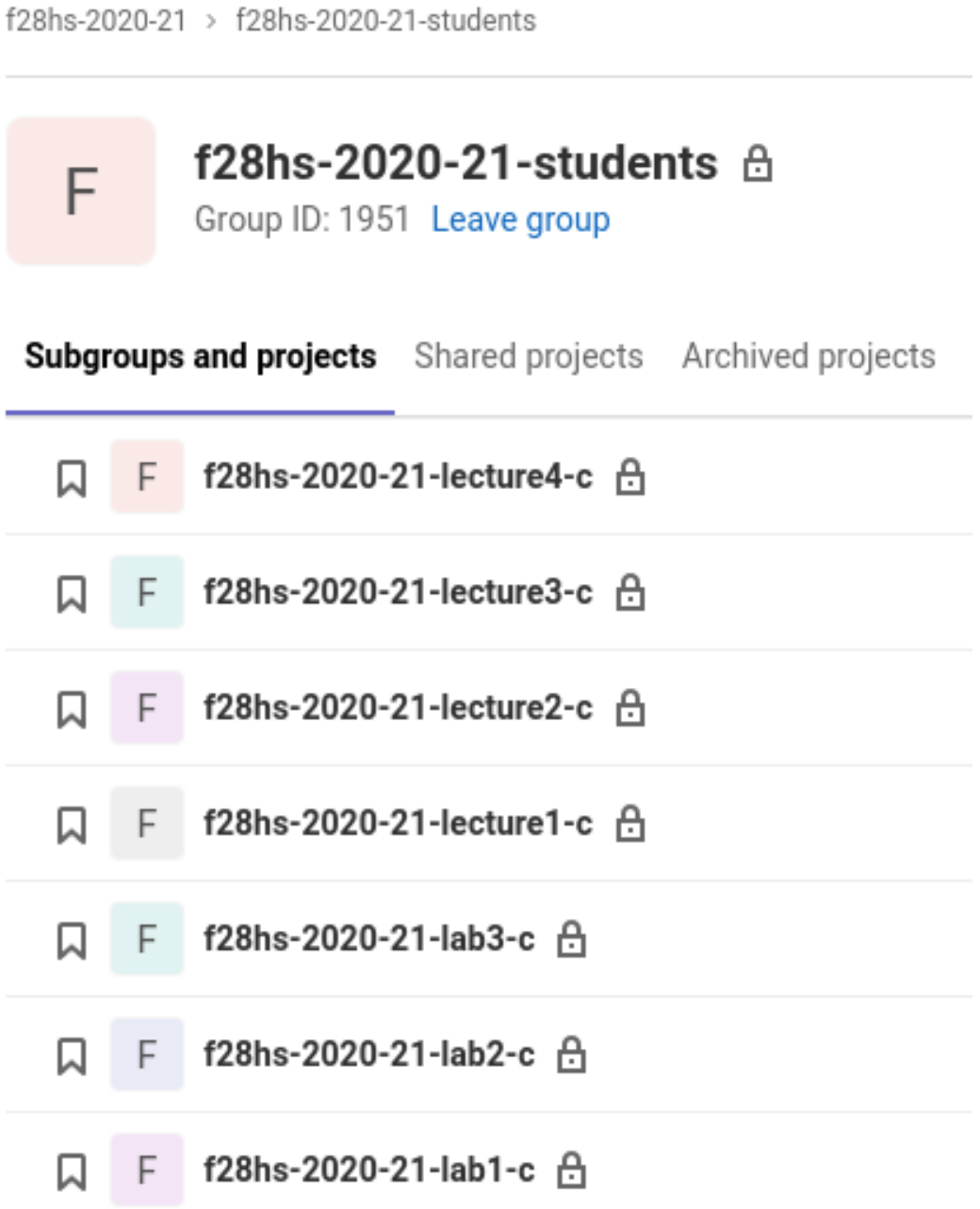
Code Clinic

- Starts in week 2 or 3
- Get help with programming questions
- Especially useful for this course
- code-clinic@macs.hw.ac.uk

- Managed by Dr Hans-Wolfgang Loidl
  - F28HS Edinburgh Course Leader
  - Will circulate details when weekly slots are timetabled
- During RBL: Microsoft Teams
- Ask on Discourse about this

GitLab Student





Software

- C programming (weeks 1-3)
  - C compiler (gcc)
  - Text editor or IDE
- Assembly programming (weeks 4-11)
  - CPUlater https://cpulator.01xz.net/
  - Text editor for submitting to GitLab
- Systems programming (weeks 6-11)
  - C compiler (gcc)
  - Text editor or IDE
- Install git

Software access

- Linux users
  - gcc and git probably already installed (if not, install them)
- Mac OSX users
  - Install gcc and git
  - Or use x2go to remotely access MACS
  - Or use the MACS Linux VM (gcc and git is installed)
  - Or use IDE (e.g. VSCode, Atom..) and graphical git software (not supported)
- Windows users
  - Use x2go to remotely access MACS
  - Or use MACS Linux VM (gcc and git is installed)
  - Or use IDE (e.g. VSCode, Atom..) and graphical git software (not supported)

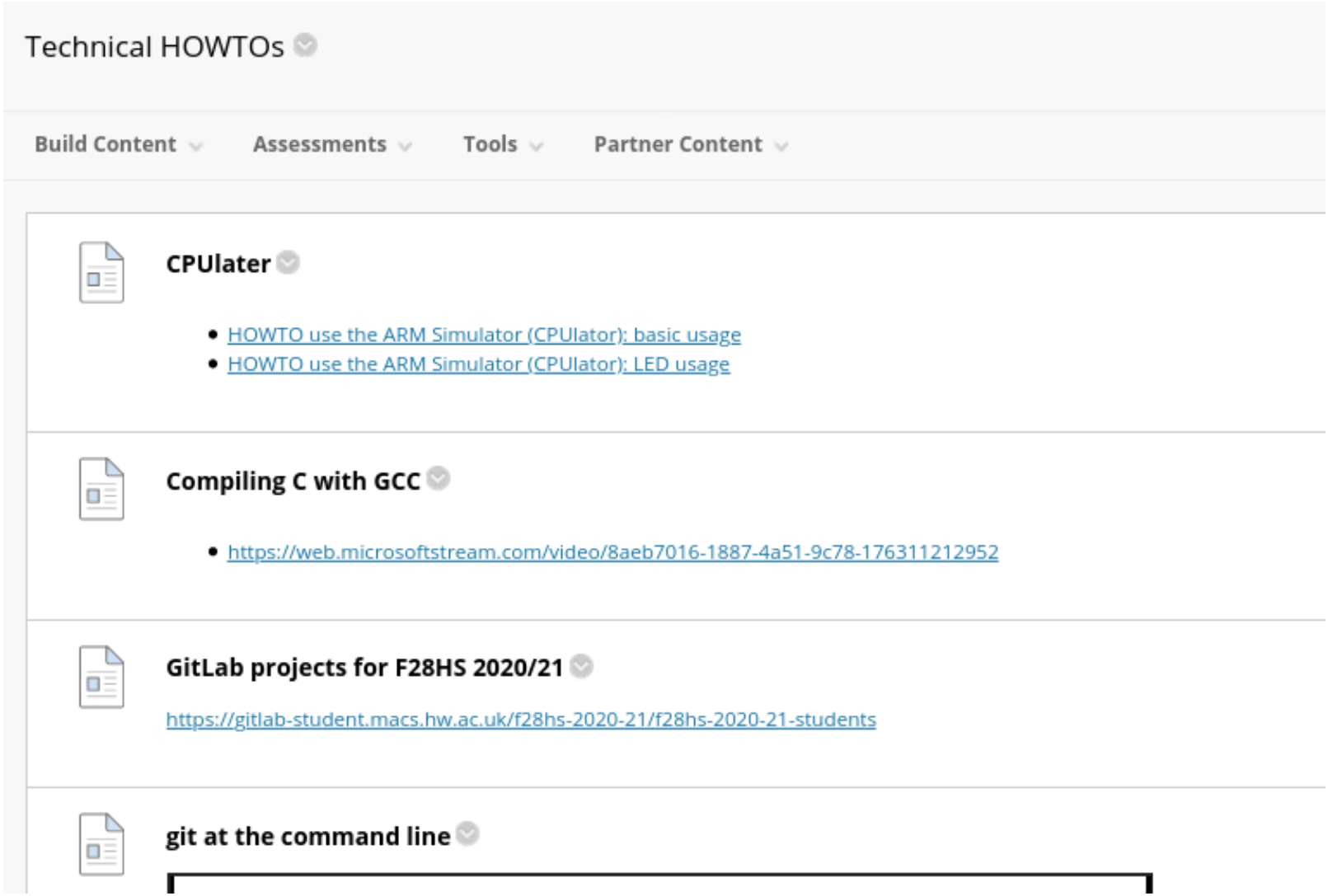
Hans-Wolfgang's Linux introduction pages

Table 1. Roadmap through the document based on your experience with Linux

	No Experience	Little Experience	Some Experience
Get Started:	the Section called *Step-by-step Login Information*	the Section called *Basic Linux Usage*	the Section called *Basic Shell Scripting Using bash*
Focus on:	the Section called *Basic Linux Usage*	the Section called *Basic Linux Usage*	the Section called *Basic Shell Scripting Using bash*
Also cover:	the Section called *Basic Shell Scripting Using bash*	the Section called *Basic Shell Scripting Using bash*	the Section called *Shell Script Examples*
Further reading:	Sobell's Linux Guide (detailed textbook on Linux, shell and much more)		
Short info:	**Linux Quick Reference Sheet (O'Reilly)**	**Linux Practical**	**Linux Command-line reference**
More detail:	**Linux Tutorial (guru99)**	**Unix Shell (software carpentry)**	**Linux Shell Scripting Tutorial (Vivek G. Gite)**
Screencasts:	**First Steps**	**Basic Linux Usage**	**Shell Scripting**
For C Programmers:	**the Section called *Practical: Edit-Compile-Run Cycle for C Programs***		

https://www.macs.hw.ac.uk/~hwloidl/Courses/LinuxIntro/t1.html

Technical HOWTOs on Vision



Industry Guest Lecture

- Company: Exterity
- Guest lecture: ***Industrial Embedded Systems Programming***
- For this company: resource-conscious programming essential
- Domain: audio signal processing within embedded systems

- Speaker hosted by Dr Hans-Wolfgang Loidl
- Invited industry speaker during Hans-Wolfgang's half of course
- Some day between week 6-11