Denne forelesningsøkten vil bli tatt opp og lagt ut i emnet i etterkant.

<u>Hvis</u> du ikke vil være med på opptaket:

	La være å delta med webkameraet ditt.
	La være å delta med mikrofonen din.
	Still spørsmål i Chat i stedet for som lyd. Hvis du ønsker kan spørsmålet også sendes privat til foreleser.

Høyskolen Kristiania

Høyskolen Kristiania

PG3401 Programmering i C for Linux

Bengt Østby

In this lecture

Høyskolen Kristiania

- Structure of the course
- Introduction to C programming
- Setting up environment

Om Bengt Østby

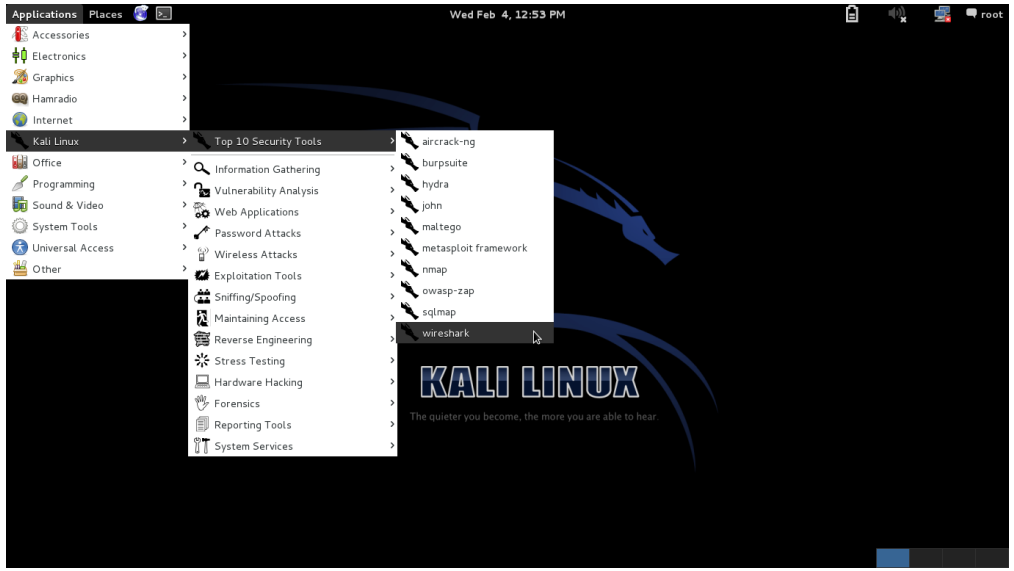
- C-programmerer og Hacker
- Windows system drivere
- 16 års erfaring fra anti-virus bransjen
- Jobbet med avansert video overvåking
- Etisk hacker og fysisk pentester

- Lead Programmer, Norman
- Enterprise Architect Security CoE, AVG
- Security Concepts Group
- Head of Offensive Security, Capgemini Cybersecurity
- Nordic lead Advanced Attack and Readiness Operations, ACN
- Foreleser Høyskolen Kristiania og USN
- Senior teknolog - Stortinget

Om Bengt Østby

Høyskolen Kristiania

- Jobber med Kali Linux på jobb



- Trener Kali Escrima på fritiden
- Og dirker låser…

Course structure

Høyskolen Kristiania

- Slides are in English, lectures will be held in Norwegian

- 12 weeks of lectures and exercises
  - Lectures and lab exercises on campus
  - Thursdays 16.15 – 20.00
  - Course is built up of 12 lectures

- Exam
  - 2-week home exam
  - Exam will be practical with several problem-solving tasks, both chosen approach to a problem, code quality – and that it actually works will count towards your final grade

- Actual CODING is essential, there is only one way to learn to code… If you only attend lectures and no exercises you will not learn to code – only some theory :-)

Lecture Plan

Høyskolen Kristiania

Lecture 1 - Introduction to the course and Linux - Why C?
Goal : -- Ability to use own Linux installation –

Lecture 2 - Intro to Tools - OS tools, Compiling tools, Debugging tools

Goal : -- Ability to write and compile a small C program–

Lecture 3 - Short intro to C, Primitive Data types, Control structures

Goal : -- Basic task solving ability –

Lecture 4 - Pointers and applications

Goal : -- Using memory better –

Lecture 5 - Strings, Arrays and Structs

Goal : -- Even more memories

Lecture Plan

Høyskolen Kristiania

Lecture 6 - I/O - terminal and files

Goal : -- A complete small single module program with functional I/O

Lecture 7 - Module-based programming, function scopes, preprocessor

Goal : -- Better organization of program into modules –

Lecture 8 - Enumerated Types, Unions, Bit operations, Threads

Goal : -- Just letting you know –

Lecture 9 - Libraries, third-party sources, advanced debugging

Goal : -- To get you not to reinvent the wheel –

Lecture 10 – Networking in C

Goal : -- Widening your scope –

Lecture 11 – Safe Programming

Goal : -- Practical and safe usage –

Lecture 12 - Repetition and Summary

Timetable of lectures and exercises

Høyskolen Kristiania

Thursday 26. August	16.15 – 20.00	Lecture 1 : Introduction to the course, and Linux
Thursday 2. September	16.15 – 20.00	Lecture 2 : Introduction to C and how to compile
Thursday 9. September	16.15 – 20.00	Lecture 3 : Datatypes and control structures
Thursday 16. September	16.15 – 20.00	Lecture 4: Pointers and applications
Thursday 23. September	16.15 – 20.00	Lecture 5: Strings, arrays and structs
Thursday 30. September	16.15 – 20.00	Lecture 6: I/O terminal and files
Thursday 7. October	16.15 – 20.00	Lecture 7: Functions and preprocessor
Thursday 14. October	16.15 – 20.00	Lecture 8: Advanced topics; unions, bits etc
Thursday 21. October	16.15 – 20.00	Lecture 9: Libraries, 3rd party and advanced debugging
Thursday 28. October	16.15 – 20.00	Lecture 10: Networking in C
Thursday 4.November	16.15 – 20.00	Lecture 11: Safe Programming
Thursday 11. November	16.15 – 20.00	Exam preparation and course summary

Exam dates not set

Books?

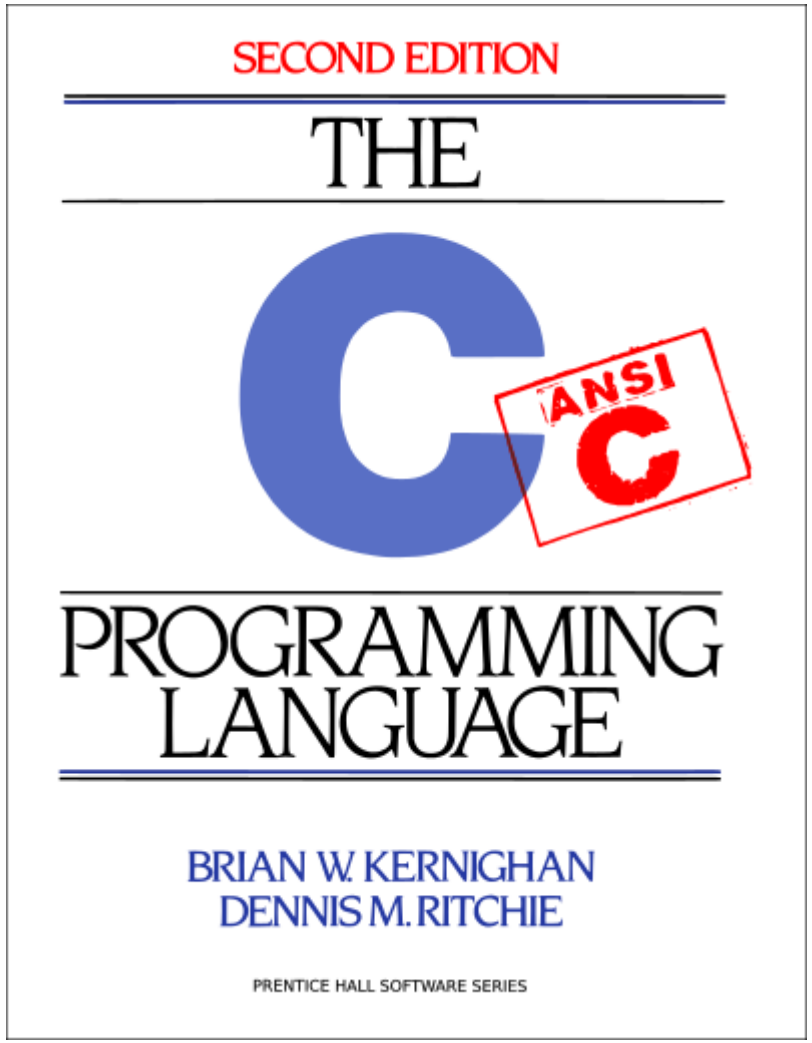
Høyskolen Kristiania

There are written thousand's of books on C.

K&R Second Edition is good

Important to support «all» embedded compilers is to learn traditional C (before C99), that is the reason for this course teaching «ANSI C»

Dennis Ritchie INVENTED the C programming language, so why read someone elses book on his language :-)



Options for environment

Høyskolen Kristiania

Course covers Linux, but the focus is on C programming.

You can run Linux in a virtual machine:

- Ubuntu
- Debian (I use Kali Linux, which is based on Debian)

You can install manually, or use preinstalled images from osboxes.org.

To run a virtual machine, you have many options:

- Virtual Box (free for students)
- VM Ware Player (free for students)
- Vm Ware Workstation

Høyskolen Kristiania

Start downloading now…

Google: “Running Debian in a vmware image” or go to:

- https://www.osboxes.org/debian/#debian-10-vmware
  (about 1Gbyte download, 5-6 Gbyte unpacked)
- Same site can also be used for Virtualbox (in other pane)

Google: “download vmware player” or:

- https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html

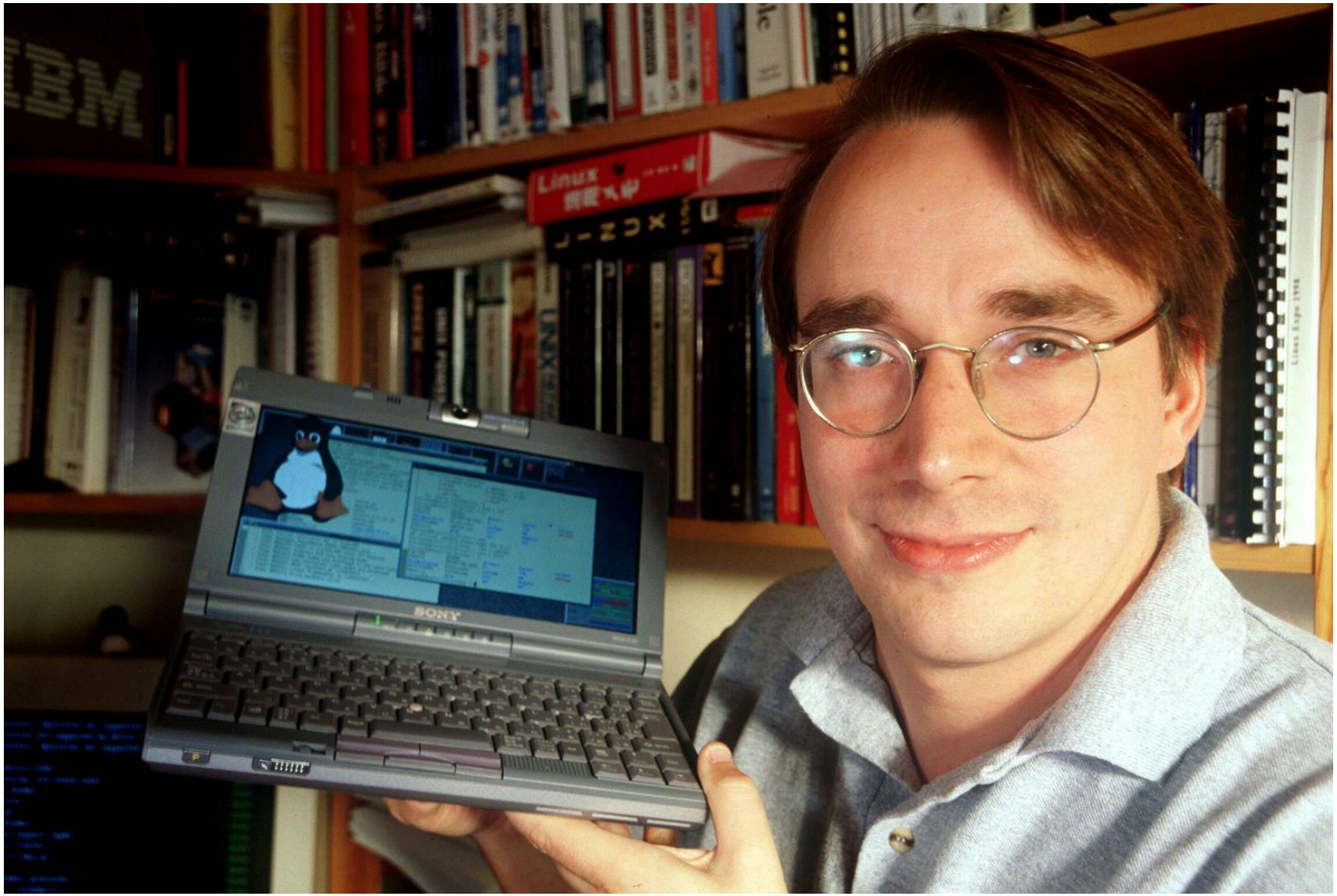
Or:

- https://www.virtualbox.org/wiki/Downloads

In the beginning, there was UNIX.

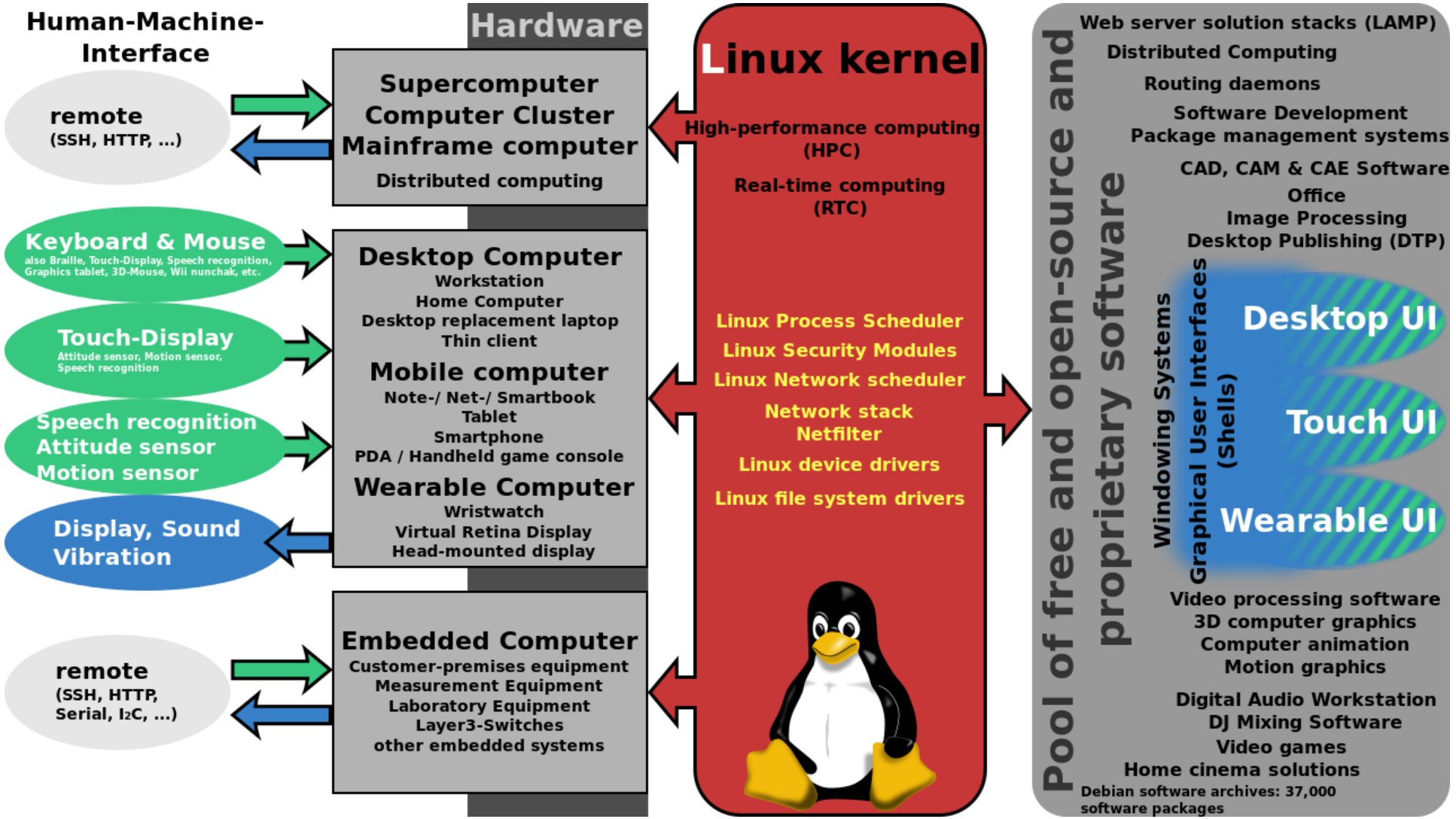
GNU is a free, cleanroom, implementation of Unix

Linux is the kernel GNU never made!



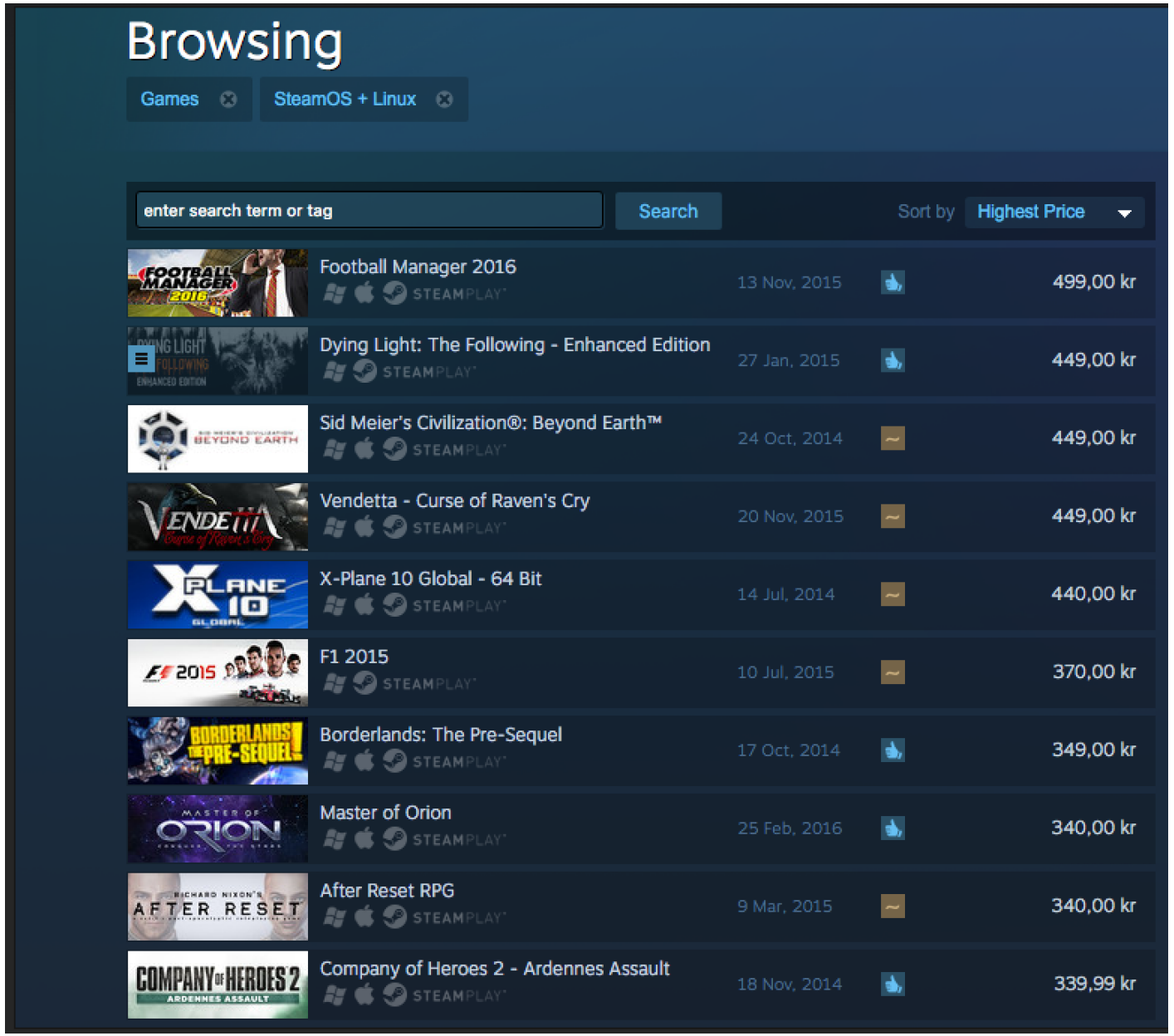
Linux is an extremely flexible OS.

Google™



Høyskolen Kristiania

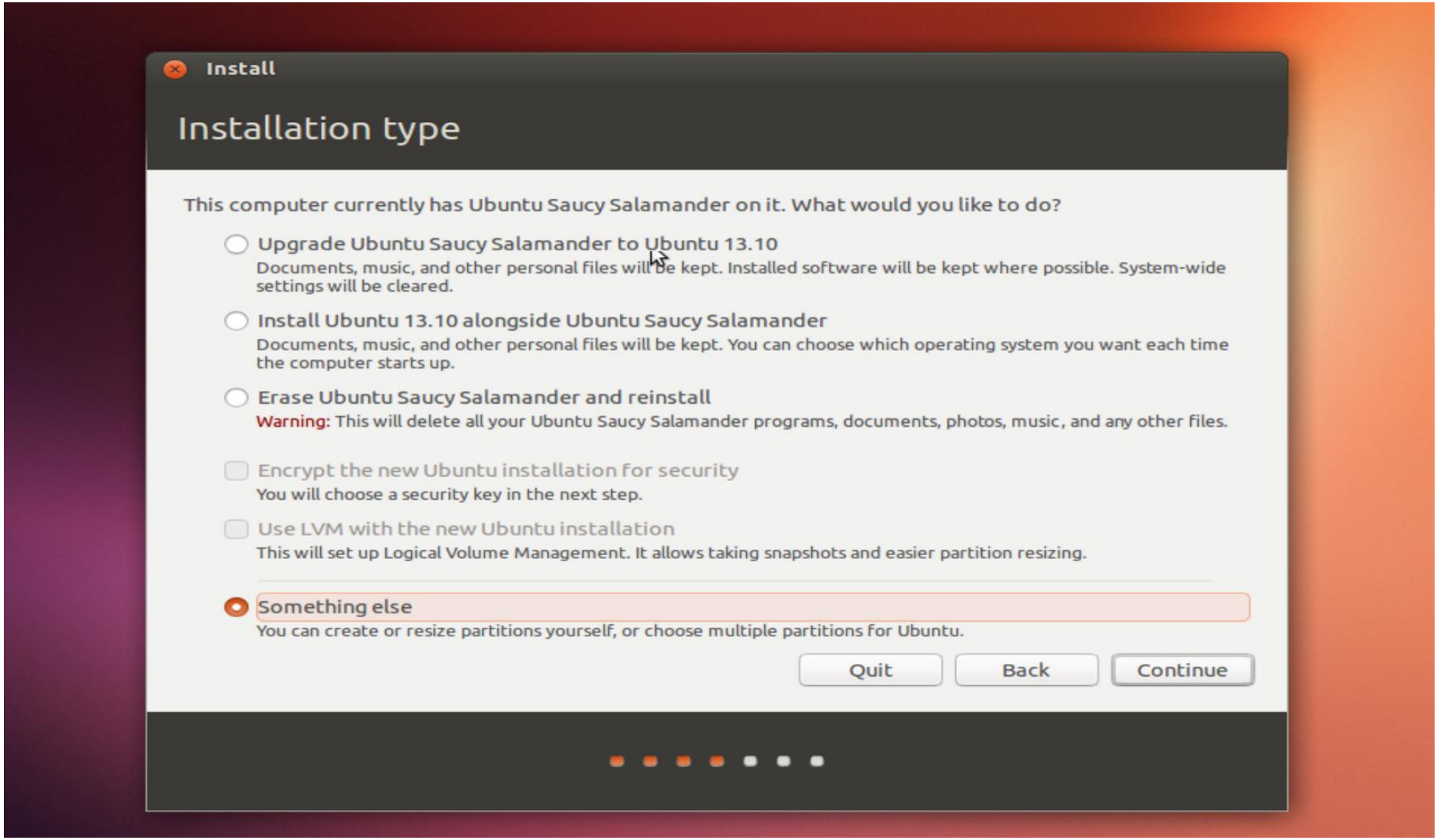
Now with games!



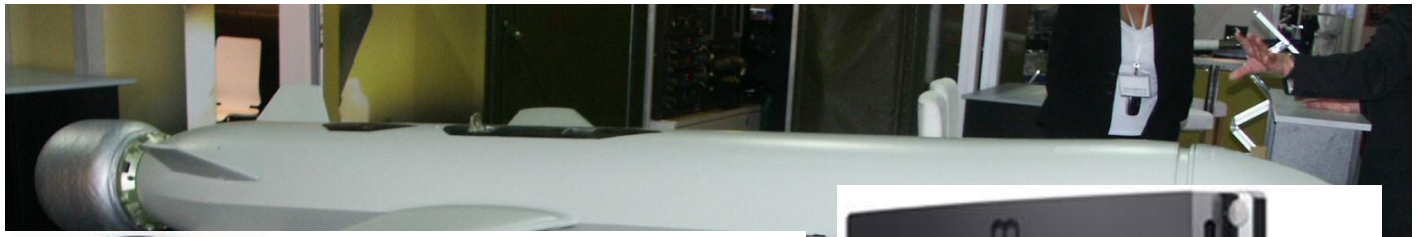
The General Public License allows redistribution, but requires you to grant the same rights to all.

Free as in Freedom

What is a “Linux Distribution”?



C is used EVERYWHERE, but you might not “C” it!



Which are the most popular programming languages in August according to TIOBE?

Høyskolen Kristiania

https://www.tiobe.com/tiobe-index//

Aug 2020	Aug 2019	Change	Programming Language	Ratings	Change
1	2	↑	C	16.98%	+1.83%
2	1	↓	Java	14.43%	-1.60%
3	3		Python	9.69%	-0.33%
4	4		C++	6.84%	+0.78%
5	5		C#	4.68%	+0.83%
6	6		Visual Basic	4.66%	+0.97%
7	7		JavaScript	2.87%	+0.62%
8	20	⇈	R	2.79%	+1.97%
9	8	↓	PHP	2.24%	+0.17%
10	10		SQL	1.46%	-0.17%
11	17	⇈	Go	1.43%	+0.45%
12	18	⇈	Swift	1.42%	+0.53%
13	19	⇈	Perl	1.11%	+0.25%
14	15	↑	Assembly language	1.04%	-0.07%
15	11	⇊	Ruby	1.03%	-0.28%
16	12	⇊	MATLAB	0.86%	-0.41%
17	16	↓	Classic Visual Basic	0.82%	-0.20%

Embedded programming is making C popular (again)…

What is C?

Høyskolen Kristiania

- Programming Language…
- Brian Kernighan and Dennis Ritchie (K&R “C”)
- Powerful programming language

- Easy to write *compilers*
- *The compiler of C is written in C!*
- *The operating system is written in C!*
- *Most high-level languages use interpreters, written in C*

- With great power comes great responsibility

The philosophy of C

Høyskolen Kristiania

- Trust the programmer
- Keep the language small and tidy
- Provide only one way to do an operation
- Make it fast, even if it’s not guaranteed to be portable
- Maintain conceptual simplicity
- Don’t prevent the programmer from doing what needs to be done.

Høyskolen Kristiania

"Programming is hard. Programming correct C and C++ is particularly hard. Indeed, both in C and certainly in C++, it is uncommon to see a screenful containing only well defined and conforming code. Why do professional programmers write code like this? ***Because most programmers do not have a deep understanding of the language they are using.*** *While they sometimes know that certain things are undefined or unspecified, they often do not know why it is so."*

Olve Mauldal

Who does C?

Høyskolen Kristiania

- Performance is priority
- Real-time system developers
- Limited resources – embedded systems
- Operating systems and drivers
- Parts of many AAA-games

- On a full and rich OS you can develop much faster in a high level language – but you can only develop what that language support!
- In C you can make everything, it will (or can) be fast and without overhead, but it is hard…
- If you are close to hardware, then you code in C

Why learn?

Høyskolen Kristiania

- To be aware of pitfalls
- Ability to develop big projects with performance priority
- Reverse engineering
- Being forced [CUDA, OpenGL, Ffmpeg, pthreads]

Some of you will go on to learn C++, which is a fully modern language based on C.

- C then C++
- C++ is based on C
- C++ does not make sense without C
- C is easier to learn
- C gives a great perspective on the Object Oriented paradigm

- Objective-C is used for mobile development, which is also a variant of C

(Don’t ever try to use C– for anything…)

This course uses Virtual Machines

Virtualization allows running multiple simultaneous OS’es on the same HW.

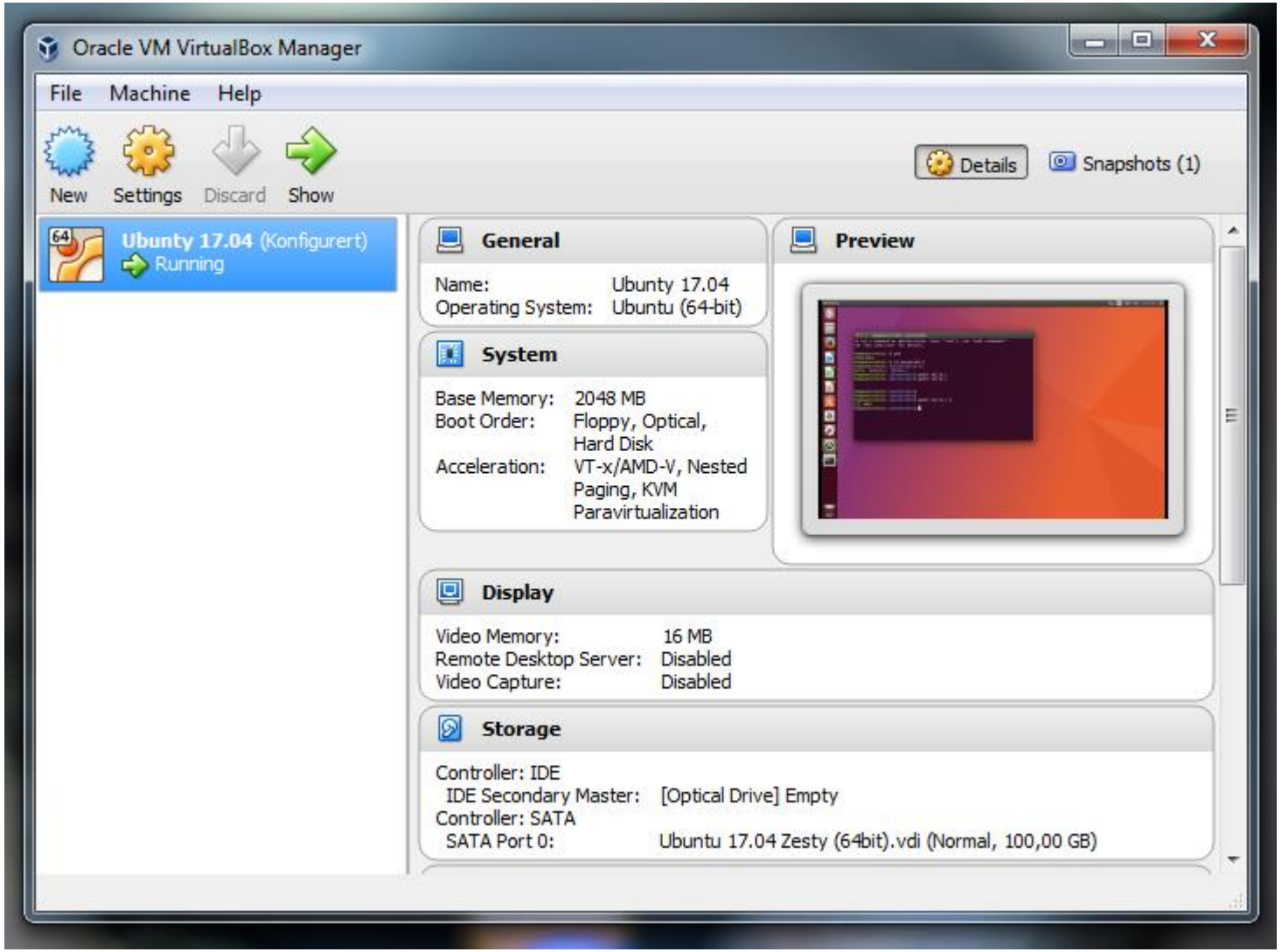
- Send finished installations around
- For testing and experiments (snapshots…)
- Popular with (anti-)virus programmers
- Consolidating servers
- The basis of Cloud computing (VM-ware, AWS, Azure…)

Virtualisation comes with its own terminology

Høyskolen Kristiania

- Host operating system
- Hypervisor
- Virtual Machine
- Guest operating system
- Snapshot

This course will use VM-ware, but cannot promise help with others kinds...



This is the luddite course!

Høyskolen Kristiania

```
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

7 packages can be updated.
0 updates are security updates.


*** System restart required ***
Last login: Mon Aug 15 11:36:10 2016 from 81.175.52.98
kjetilr@oslo:~$ ls
kjetilr@oslo:~$ ls /etc/
X11                     console-setup    fstab            inputrc          libaudit.conf   mdadm
acpi                    cracklib         fuse.conf        insserv          libnl-3         mime.types
adduser.conf            cron.d           gai.conf         insserv.conf     lintianrc       mke2fs.conf
alternatives            cron.daily       groff            insserv.conf.d   locale.alias    modprobe.d
apm                     cron.hourly      group            iproute2         locale.gen      modules
apparmor                cron.monthly     group-           iscsi            localtime       modules-load.d
apparmor.d              cron.weekly      grub.d           issue            logcheck        mtab
apport                  crontab          gshadow          issue.net        login.defs      nanorc
apt                     crypttab         gshadow-         kbd              logrotate.conf  network
at.deny                 dbus-1           gss              kernel           logrotate.d     networks
bash.bashrc             debconf.conf     hdparm.conf      kernel-img.conf  lsb-release     newt
bash_completion         debian_version   host.conf        krb5.conf        ltrace.conf     nsswitch.conf
bash_completion.d       default          hostname         krb5.conf.old    lvm             ntp.conf
bindresvport.blacklist  deluser.conf     hosts            krb5.keytab      machine-id      opt
binfmt.d                depmod.d         hosts.allow      ld.so.cache      magic           os-release
byobu                   dhcp             hosts.deny       ld.so.conf       magic.mime      overlayroot.conf
ca-certificates         dpkg             init             ld.so.conf.d     mailcap         pam.conf
ca-certificates.conf    environment      init.d           ldap             mailcap.order   pam.d
calendar                fonts            initramfs-tools  legal            manpath.config  passwd
kjetilr@oslo:~$
```

We will learn to be “old-school”

Høyskolen Kristiania

- For those familiar with modern IDE, coding and compiling on command-prompt on Linux will feel like exchanging a modern drill with a box of screwdrivers
- But you will all thank me when you work with embedded systems (or kernel mode drivers)
- Debugging crashes on a machine without a keyboard or monitor, or a system that crashes during boot sequence - is extremely difficult, but possible as long as we stay «old-school»

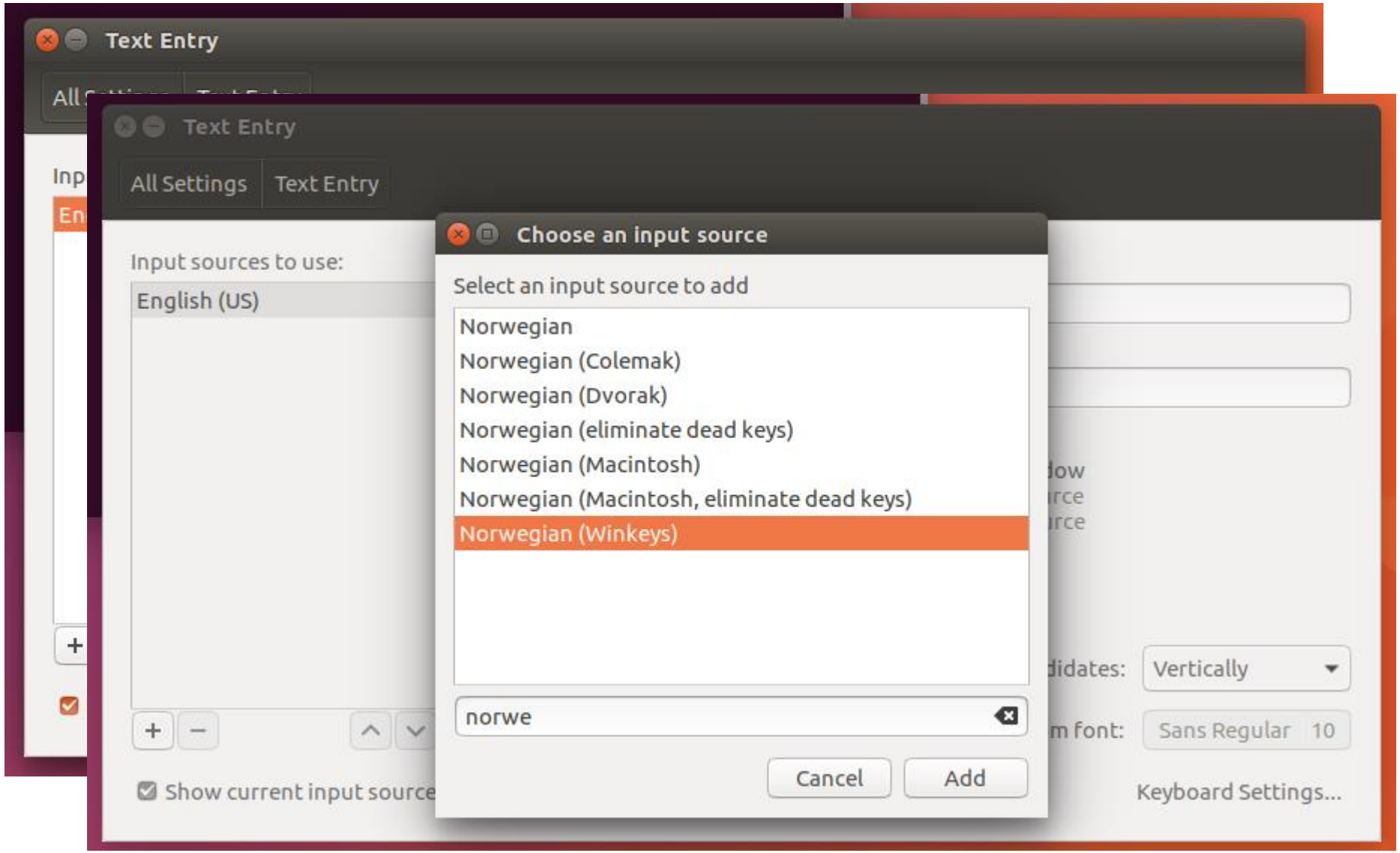
Today’s assignment

Høyskolen Kristiania

- Install Linux
- Run Linux
- Make account with new password
- Delete old account
- Configure as you like it (keyboard layout??)
- Surf the web, check mail
- Try Gedit

Set NO as preferred keyboard layout

Høyskolen Kristiania



Høyskolen Kristiania

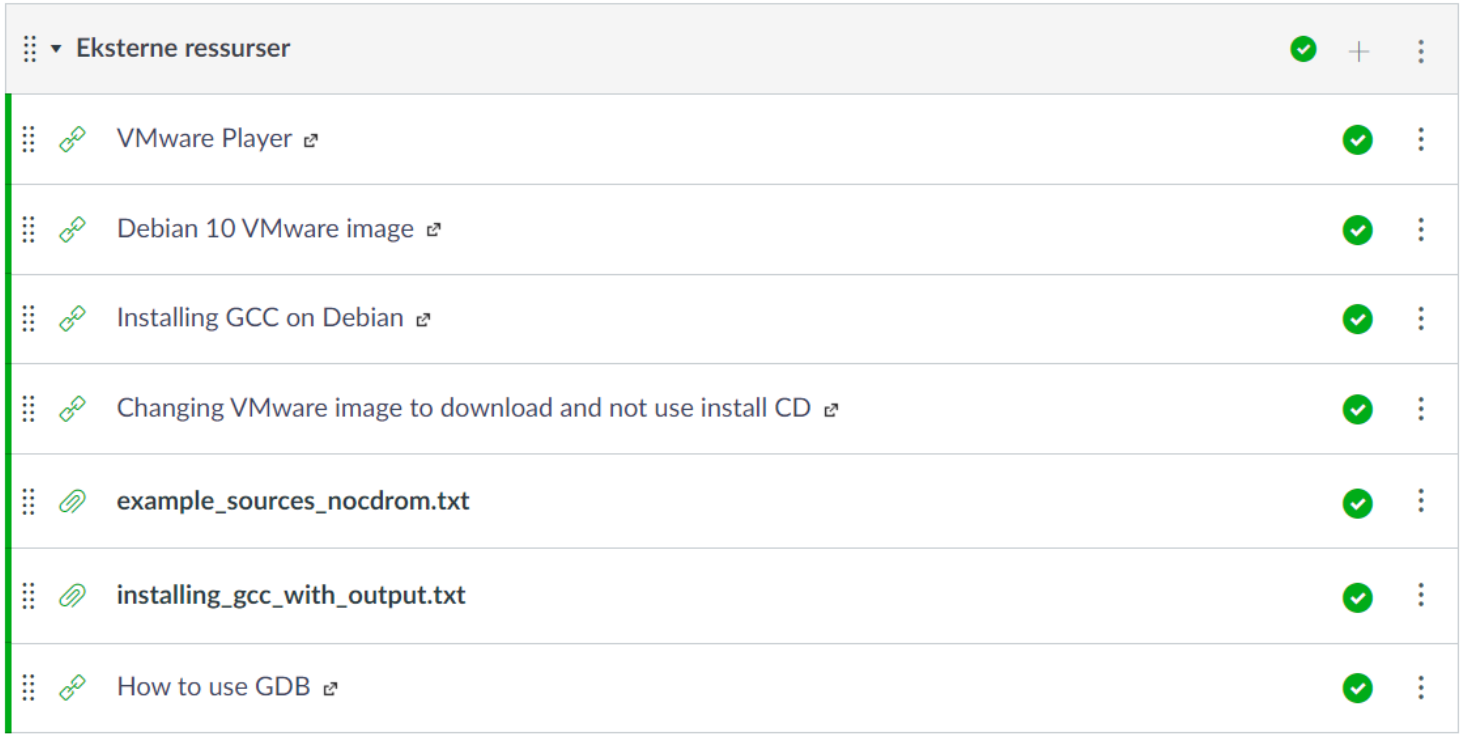
Practical

Practical – Installing:

Høyskolen Kristiania

- **Install VmWare Player**
- **Install Debian 10 image**
- **Install GCC**

- **Linker ligger i Canvas:**



Practical – Linux :

Høyskolen Kristiania

wget http://www.eastwillsecurity.com/pg3401/linux.txt

Try to read the file

Open a new tab in terminal

Follow the instructions from the file

Lab exercises

Høyskolen Kristiania

Help each other becoming used to Linux. Some of you might have used it before, some are new to this OS. Make sure you all help each other – this is one of the most difficult courses you will take at this school, everyone of you will need help from the class! :-) You must attend Campus trainings, trying to learn it all your self will most often fail, hard…

After next week everyone in the class MUST have a running Linux environment and be able to edit and compile code!