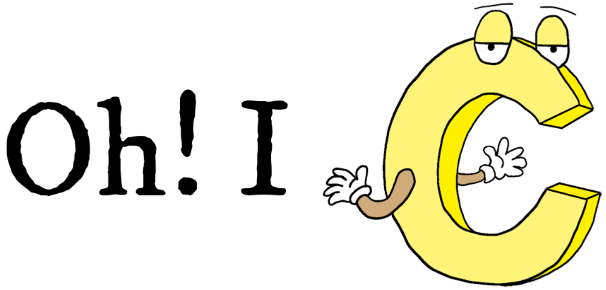
C Programming I (FDY400)

Andreas Heinz
FYD400 - HT 2019

```
void loop()
{
    //MCU Task
    for(NUM_FN_TASK_CNT = 0; ((NUM_FN_TAS
    {
        if ((millis() - fn[NUM_FN_TASK_C
        {
            fn[NUM_FN_TASK_CNT].time_cnt =
```



General Information I

- **Course instructor:**

  Andreas Heinz, TF, 772-3430 (andreas.heinz@chalmers.se)
  F8008 – Forskarhus Fysik

- **Course teachers:**

  Vitali Zhaunerchyk (vitali.zhaunerchyk@physik.gu.se)

  Adrian Rodriguez Palomo (adrian.rodriguez@chalmers.se)

  Shahnawaz Ahmed (shahnawaz.ahmed@chalmers.se)

- **Course web page: FDY 400 via CANVAS canvas.gu.se**

  => you need to log in to get access to the contents!

- **You still need to fill in the form today! Registration alone is not sufficient!**

General Information II

If you are registered, you obtain your login for the CANVAS web page of the course in LADOK.

If you need a CID/CDKS (a computer account to use the computers in the lab), please contact Pernilla Larsson (administrator).

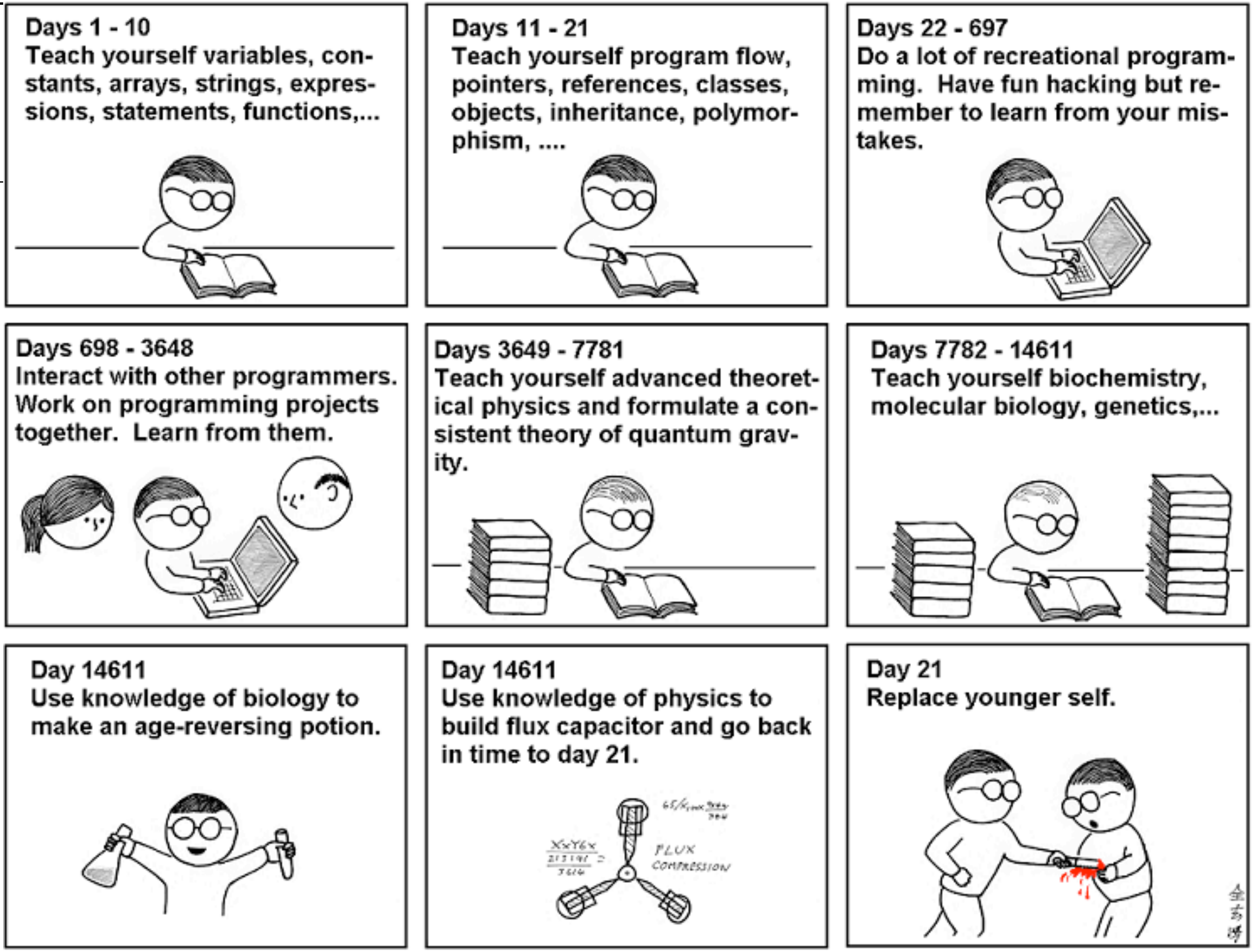
Course Schedule and Examination

- Course schedule:
  - 6 lectures (including this introduction) in FB-salen (18:00 – 20:00)
  - 9 practical exercises (includes exam) in F7105A (18:00 – 22:00)
- Examination: solve and present tasks given in teams of 2 students;
- End date: Wednesday, October 30, 2019
- Course literature: ”Vägen till C” by Bilting & Skansholm (Cremona)
- Many other books are on the market! A lot of resources are available online!

Lectures

- Lectures
  - No prior knowledge on computers and programming required.
  - They provide only a brief introduction to programming.
  - Support for learning and examples
- NO substitute for practical work and reading!

> In theory there is no difference between theory and practice; in practice there is.
>
> Yogi Berra
>
> Everything is practice.
>
> Pele


Days 1 - 10
Teach yourself variables, constants, arrays, strings, expressions, statements, functions,...
Days 11 - 21
Teach yourself program flow, pointers, references, classes, objects, inheritance, polymorphism, ...
Days 22 - 697
Do a lot of recreational programming. Have fun hacking but remember to learn from your mistakes.
Days 698 - 3648
Interact with other programmers. Work on programming projects together. Learn from them.
Days 3649 - 7781
Teach yourself advanced theoretical physics and formulate a consistent theory of quantum gravity.
Days 7782 - 14611
Teach yourself biochemistry, molecular biology, genetics,...
Day 14611
Use knowledge of biology to make an age-reversing potion.
Day 14611
Use knowledge of physics to build flux capacitor and go back in time to day 21.
FLUX COMPRESSION
Day 21
Replace younger self.

abstrusegoose.com/249

Works similar for C

As far as I know, this is the easiest way to "Teach Yourself C++ in 21 Days".

Practical Work

Exercises and examination:

The most important part of this course!

- (Limited) help for solving problems.
- In ET and labbet: all examination is done here (personal laptops are allowed)
- Do not come at the last moment…
- Requirements for passing a problem: working program, the demonstration of good programming practice, an understanding of the material (oral), proper I/O, comments
- Many tasks can/need to be done (started) at home!
  - www.ni.com (National Instruments) demo
  - Cost-free alternatives to LabCVI, e.g. GNU C compiler, X code, etc.
- Attend several times!

Grading - Summary

Complete information in the compendium available.

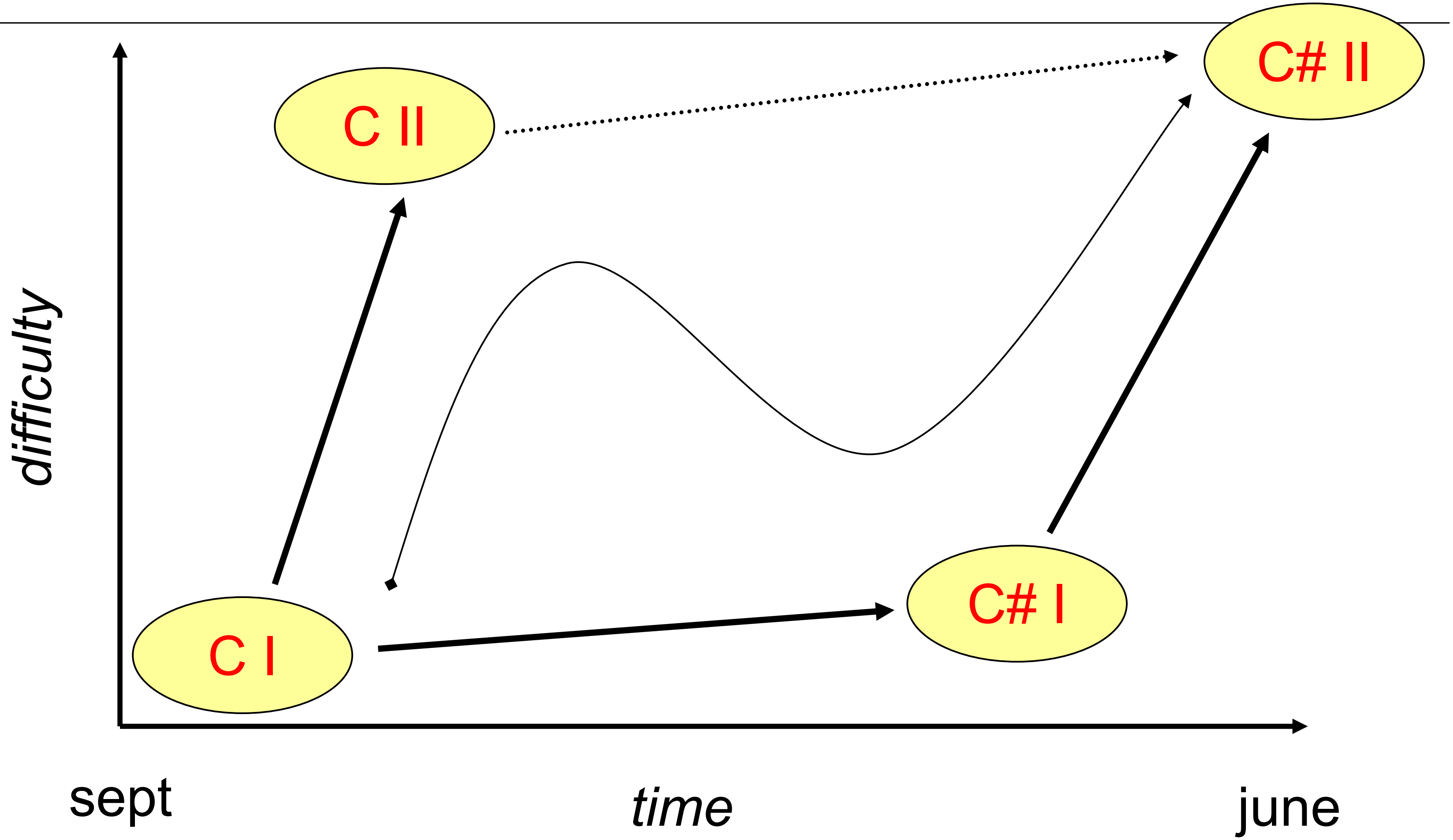
No hand-ins! All problems need to be presented during the exercises.

You need to have at least one solved problem per sub-chapter to pass the course!

Grade G:
You need to have at least 30 points.

Grade VG:
You need to have at least 40 points.

The Big Picture



Preliminary Lecture Plan

Introduction:

General course information, registration for a computer account and a key card

Lecture 1:

Computers and programming, history and development of C, programming [Chapter 1, 2]

Lecture 2:

Nomenclature, characters, types, variables, standard functions, preprocessor commands demonstration of LabWindows CVI (GUI, controls, debugging, modules), expressions, operators, arithmetic, control structures [Chapter 3, 4, 9, 10]

Lecture 3:

Expressions, functions, storage classes [Chapters 5,6]

Lecture 4:

Pointer, fields, records, lists, unions, type definitions [Chapters 7,8]

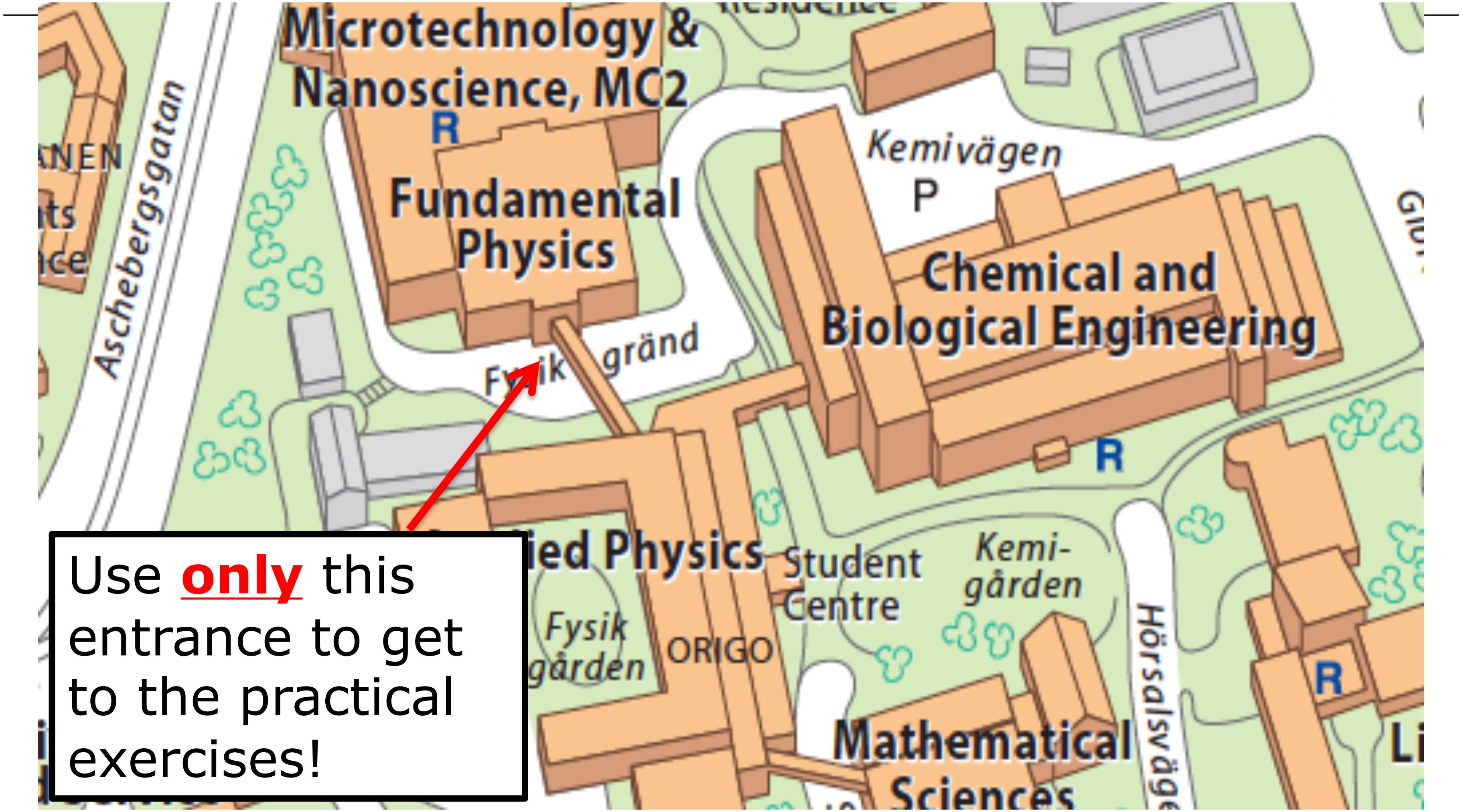
Lecture 5:

File management, string handling, modern "C" [Chapter 10]

Please Fill in the List!

- **Important:**
  - **Please fill in the list! Please write READABLE!**
  - **name, personnummer, and if you need an access card to ET labbet.**
- **Questions about the course?**

Access for Exercises



Use **only** this entrance to get to the practical exercises!