TODAY ○ SYLLABUS + COURSE METHODOLOGY

○ TEACHING MAKE UP ⟶ WED pm?

↳ TUE APRIL 17
→ TUE MAY 1 maybe THU MAY 3

Syllabus → TEXTBOOK
→ GRADING
→ HOMEWORK ⟶ 2/WK
EXAMS ⟶ presentations
interview

↓

[no UB]

[INDEX CARDS]

↳ TODAY: NAME
SID
YEAR
(GOALS)
↳ WHAT SHOULD I KNOW?
OP. SYS.
PROG B/G

caveat emptor: I CANNOT HELP W/ TECHNICAL COMP. ISSUES

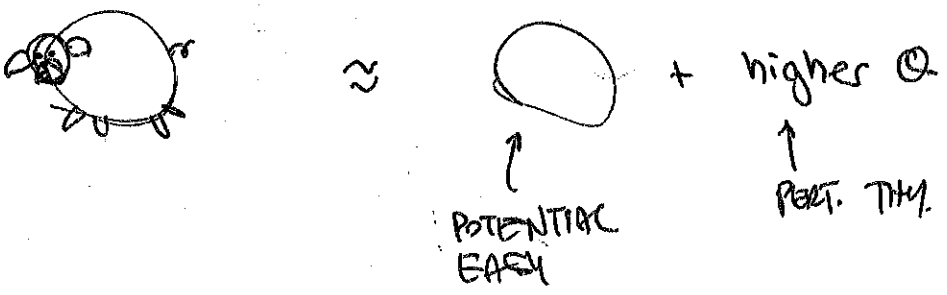
Computational physics

RELATIVELY NEW

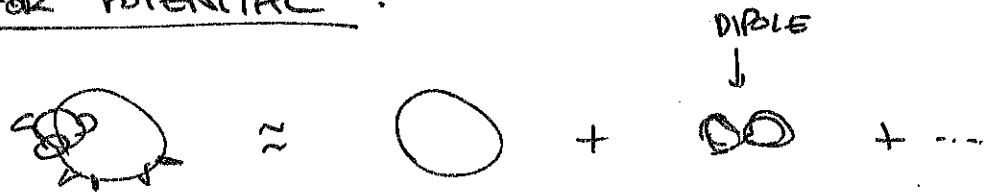
↑ OLD ... "never" needed computers...

Why? 1. SPHERICAL COWS (thy)
2. GRAPH PAPER (exp.)

THEORY:



for POTENTIAL :



$\Phi_{tot} = \Phi_{mono} + \Phi_{di} + \cdots$

nice simple SUM
B/C THE DYNAMICS ARE LINEAR

read: "nice"

LIFE ISN'T NICE → non-linear

either do a clever expension

⋮

OR: COMPUTER

EXPERIMENT : big science → big data
(why tech companies ♥ physics)

graph paper → spreadsheet → csv, ...

LARGE MULTI D/M DATA
(meas, uncertainty
spectrum of a pulse
momenta of 100s
of particles ..

gotta process it

BONUS : RANDOM #'s ← computers are great for this

→ in physics → QUANTUM
→ STATISTICAL

so we can simulate models that are based on inherent randomness.

↑ eg determine collective behavior of zillions of electrons in a material.

---

class goals : ~~have a break each~~

EACH CLASS :

- ~1/2 THEORY
- ~1/2 APPLICATION (together on JUPYTER) ← feel free to bring laptop

- BREAK
- occasional index card ← why: active feedback

2 HW / wk : short (2 days) + long (7 days)
↑ QUICK ↑ SUBMIT on github

this course is not:

- ALGORITHMS for CS
  - ↳ not "numerical recipes"
    - understand basic recipes & how / why they work
      - ↳ & when they don't!

- PROGRAMMING → you should know enough to get by
  - We will use Python 3 + Jupyter.

↑ you will never be required to "code code" in this class.

... use whatever refs you want (eg: GOOGLE)

you will be responsible for understanding.

WE'RE HACKING, not PROGRAMMING.
↑ quick, readable, ...

LANGUAGE : Python 3 + Jupyter + GitHub

↑ POPULAR EASY USEFUL

↑ READABLE

sharing code & HW submit

these 3 skills are essential for

(a) modern physics

(b) software industry

ACADEMIC INTEGRITY :

- collaborate
- cite

} we will be very strict w/ plagiarism

HUMAN THOUGHT is often essential

eg: $\int_0^1 \left[f_1(x) - f_2(x) \right] \Theta(f_1(x) - f_2(x))$

$f_1(x) = (\text{small}) x^2 + (\text{small positive})$

$f_2(x) = (\text{big}) x^2$

SIMPLE NUMERICAL INTEGRATION WAS SLOW, UNRELIABLE.

why?

JUPYTER

terminal: `jupyter notebook`

opens browser

`NEW > Python 3 notebook`

EDIT TITLE

~~WRITE~~ DROPDOWN MENU → `Markdown`

WRITE, eg:

```
# our first jupyter
Name
class + Term
```

`Cell > Run cell` or `shift + enter`

creates new cell below in code mode.

try some math.

save + check.

└ close window, terminal: `cntrl+c`, y

`GITHUB` → see 1