1.1 Radioactive Decay

- ^{235}U (uranium nucleus), small probability for decaying into two nuclei
- $Nu(t)$ is the number of uranium nuclei present

$$\frac{dNu}{dt} = -\frac{Nu}{\tau}$$

- t is time
- τ is "time constant"
- use this ODE $\Rightarrow$ $Nu = Nu(0)e^{-t/\tau}$
- $Nu(0)$ is initial number of Nucleus at $t=0$

1.2 Numerical Approach

- Use an initial value problem, obtaining Nu as a function of t. give Nu at $t=0$
- Taylor expansion

$$Nu(\Delta t) = Nu(0) + \frac{dNu}{dt}\Delta t + \frac{1}{2}\frac{d^2Nu}{dt^2}(\Delta t)^2 + \dots$$

$$\frac{dNu}{dt} = \frac{Nu(t+\Delta t) - Nu(t)}{\Delta t}$$

- Since Δt is made to be small, we can expect negligable error
- Final approximation

$$Nu(t+\Delta t) \approx Nu(t) - \frac{Nu(t)}{\tau}\Delta t$$

- we can estimate the value at a later time (Δt)
- Euler Method

1.3 Design and construction of a program

-pseudocode is language to be interpreted

- Think!

-build an outline for the code before jumping in

Overall structure

1. Declare necessary variables & arrays
2. initialize variables
3. do the calculation
4. store the results

Declare the "Function" and arrays

Call the variables with different functions

Initialize sets the initial values

Calculate performs Euler method

Store is like testing and calling

- Variable lists must correspond
- Results best expressed graphically

1.4 Testing your program

Does the output look reasonable?

-should have a rough idea about the results

Exact Results?

Compare program to known values

Make sure the same output for different nsteps

1.5 Numerical Considerations

- round-off errors present with finite # of digits
- calculations should be repeated with diff values
- make sure it converges to a curve as nsteps ↑
- small fraction of y for diff nsteps
- no one best method for solving ODE