

review of the first 21 lectures

- 1 Exam on Friday 13 October, at 11AM
 - in class or take home?
- 2 Review Questions
 - vectorization
 - symbolic computing
 - visualization
 - connectivity
- 3 Midterm Exam of Spring 2022
 - vectorization
 - symbolic, numeric, algorithmic differentiation
 - parallel computation with Julia
 - interfaces

MCS 507 Lecture 22
Mathematical, Statistical and Scientific Software
Jan Verschelde, 11 October 2023

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- symbolic computing
- visualization
- connectivity

3 Midterm Exam of Spring 2022

- vectorization
- symbolic, numeric, algorithmic differentiation
- parallel computation with Julia
- interfaces

exam policy

Friday 13 October is our midterm exam, which could be either

- An in-class conventional exam, open book and notes, the exam can be done at home;

or

- A take-home exam due on Monday 16 October at 11AM which must be solved individually, but will need the use of software.

The decision to do in-class or take-home can be postponed till 11:49AM on Friday 13 October, just do not upload your solution into gradescope if you want to do the take-home version.

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- **vectorization**
- symbolic computing
- visualization
- connectivity

3 Midterm Exam of Spring 2022

- vectorization
- symbolic, numeric, algorithmic differentiation
- parallel computation with Julia
- interfaces

vectorization

Consider the composite trapezoidal rule:

$$\frac{h}{2}f(a) + h \sum_{i=1}^{n-1} f(a + ih) + \frac{h}{2}f(b), \quad h = \frac{b-a}{n}$$

to approximate $\int_a^b f(x)dx$.

Assume $f(x)$ is a vectorized function, e.g. $f = \cos$ of `numpy`.

Define the vectorization of the trapezoidal rule.

Take home question: Write a script to compare the relative speeds of the original trapezoidal rule with its vectorized version.

Take n sufficiently large to observe a significant difference.

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- **symbolic computing**
- visualization
- connectivity

3 Midterm Exam of Spring 2022

- vectorization
- symbolic, numeric, algorithmic differentiation
- parallel computation with Julia
- interfaces

interpolating polynomials

Consider a list of tuples (x_i, y_i) , for $i = 0, 1, \dots, n$ with $x_i \neq x_j$ for all $i \neq j$. The Lagrange polynomials are defined as

$$L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j}$$

and $p(x) = \sum_{i=0}^n y_i L_i(x)$ is an interpolating polynomial: $p(x_i) = y_i$, for $i = 0, 1, \dots, n$. Write Python code using `sympy` to define $L_i(x)$ and p for a given list of tuples.

Take home question: Write a module for Lagrange interpolation with `sympy.Poly` objects. As test function, generate a random polynomial q of degree n with integer coefficients and let the list of $n + 1$ integer points $(k, q(k))$ for $k = 0, 1, \dots, n$. The test verifies that $p \equiv q$.

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- symbolic computing
- **visualization**
- connectivity

3 Midterm Exam of Spring 2022

- vectorization
- symbolic, numeric, algorithmic differentiation
- parallel computation with Julia
- interfaces

visualization

Consider the complex roots of a polynomial as points in the plane, i.e.: the root $a + ib$ corresponds to the point (a, b) .

For a polynomial p , the theorem of Gauss Lukas states that the roots of the derivative p' belong to the convex hull spanned by the roots of p .

Use `numpy`, `scipy`, and `matplotlib` or alternatively SageMath, to illustrate the Gauss-Lukas theorem.

Take home question: Design a graphical user interface with Tkinter to illustrate the Gauss-Lukas theorem.

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- symbolic computing
- visualization
- **connectivity**

3 Midterm Exam of Spring 2022

- vectorization
- symbolic, numeric, algorithmic differentiation
- parallel computation with Julia
- interfaces

connectivity

Consider two Python interpreters A and B :

- 1 A has no capability for root finding.
- 2 B has access to `numpy` and its `roots` function.

Describe a mechanism for A to pass the coefficients of a polynomial to B and then retrieve the roots from B .

Take home question: Write a script which prints a form for the user to enter the coefficients of a polynomial.

After the user submits the form, the script prints the roots into the browser window.

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- symbolic computing
- visualization
- connectivity

3 Midterm Exam of Spring 2022

- **vectorization**
- symbolic, numeric, algorithmic differentiation
- parallel computation with Julia
- interfaces

vectorization

Consider on input a list X of n floating-points numbers.
The formulas to compute the mean μ and standard deviation σ are

$$\mu = \frac{1}{n} \sum_{k=1}^n X_k \quad \text{and} \quad \sigma = \sqrt{\frac{1}{n} \sum_{k=1}^n (X_k - \mu)^2}$$

Describe the vectorization of the computation of μ and σ .

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- symbolic computing
- visualization
- connectivity

3 Midterm Exam of Spring 2022

- vectorization
- **symbolic, numeric, algorithmic differentiation**
- parallel computation with Julia
- interfaces

differentiation

Consider the function

$$g(z) = \frac{\exp(z)}{(\cos(z))^3 + (\sin(z))^3}.$$

Suppose we need to compute the first 10 derivatives of $g(z)$ at 1.

Compare the application of symbolic, numeric, algorithmic differentiation to this problem.

For each of the three methods, list the advantages and disadvantages.

Which is the preferable method for this problem?

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- symbolic computing
- visualization
- connectivity

3 Midterm Exam of Spring 2022

- vectorization
- symbolic, numeric, algorithmic differentiation
- **parallel computation with Julia**
- interfaces

parallel computation with Julia

Consider the composite trapezoidal rule:

$$\frac{h}{2}f(a) + h \sum_{i=1}^{n-1} f(a + ih) + \frac{h}{2}f(b), \quad h = \frac{b-a}{n}$$

to approximate $\int_a^b f(x) dx$.

Describe the setup in Julia of a distributed computation of this rule.

- 1 What data will be distributed among the processors?
- 2 Indicate the Julia construction that you will apply.
- 3 Outline a test function to demonstrate the speedup.

review of the first 21 lectures

1 Exam on Friday 13 October, at 11AM

- in class or take home?

2 Review Questions

- vectorization
- symbolic computing
- visualization
- connectivity

3 Midterm Exam of Spring 2022

- vectorization
- symbolic, numeric, algorithmic differentiation
- parallel computation with Julia
- **interfaces**

interfaces

Consider two Python interpreters A and B :

- 1 A has no capability for solving ordinary differential equations.
- 2 B has access to `scipy` to solve ordinary differential equations.

Describe a mechanism for A to pass the definition of an ODE to B and then to retrieve the solution from B .