

Welcome to MCS 260

About the Course

Content
Expectations

Algorithms

What is Computer Science?
Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

- 1 About the Course
Content
Expectations
- 2 Algorithms
What is Computer Science?
Introduction to Python
- 3 Mathematical Software
The three big M's
Open Source: SAGE
- 4 Summary + Assignments

MCS 260 Lecture 1
Introduction to Computer Science
Jan Vershelde, 11 January 2010

Introduction to Computer Science

- 1 About the Course
Content
Expectations
- 2 Algorithms
What is Computer Science?
Introduction to Python
- 3 Mathematical Software
The three big M's
Open Source: SAGE
- 4 Summary + Assignments

Content of the Course

Text Books

The design of the course follows

- J. Glenn Brookshear: *Computer Science. An Overview.* 9th edition, Addison Wesley, 2007.
- Bradley N. Miller and David L. Ranum: *Python Programming in Context.* Jones and Bartlett, 2009.

Two text books also good for the sequel to MCS 260:
MCS 275: Programming Tools and File Management.

Another sequel to MCS 260 is MCS 320:
Introduction to Symbolic Computation.

Content of the Course

Text Books

About the Course

Content

Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's

Open Source: SAGE

Summary + Assignments

The design of the course follows

- J. Glenn Brookshear: *Computer Science. An Overview.* 9th edition, Addison Wesley, 2007.
- Bradley N. Miller and David L. Ranum: *Python Programming in Context.* Jones and Bartlett, 2009.

Two text books also good for the sequel to MCS 260:
MCS 275: Programming Tools and File Management.

Another sequel to MCS 260 is MCS 320:
Introduction to Symbolic Computation.

Content of the Course

Text Books

The design of the course follows

- J. Glenn Brookshear: *Computer Science. An Overview.* 9th edition, Addison Wesley, 2007.
- Bradley N. Miller and David L. Ranum: *Python Programming in Context.* Jones and Bartlett, 2009.

Two text books also good for the sequel to MCS 260:
MCS 275: Programming Tools and File Management.

Another sequel to MCS 260 is MCS 320:
Introduction to Symbolic Computation.

Revised Catalog Description for

mcs 260

still in progress...

About the Course

Content

Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's

Open Source: SAGE

Summary + Assignments

Computer literacy, number systems, concepts of operating systems, storage, files, databases, logic gates, circuits, networks, internet.

Introduction to programming in Python, variables, assignments, functions, objects, defining algorithms to solve problems, software design and testing.

Basics of graphical user interfaces and web servers.

Computer Science.

An Overview

J. Glenn Brookshear

- offers a solid introduction to Computer Science
 - hardware: how a computer works
 - software: how to use a computer
- computer literacy
 - understand basic terminology
 - overview of a discipline
- coverage:
 - explicitly several essential materials
 - use implicitly as a general reference

Python Programming in Context

Bradley N. Miller and David L. Ranum

- concise treatment of a broad range of topics
- we will cover
 - 1 in first 18 lectures (till 1st midterm):
data structures, control constructs, functions
→ programming in the small (chapters 1 to 5)
 - 2 lectures 21 to 36 (between midterms):
modules, manipulating files, classes and objects
→ programming in the large (chapters 10, 11, 12)
 - 3 video game development (chapter 13)

Introduction to Computer Science

- 1 About the Course
 - Content
 - Expectations
- 2 Algorithms
 - What is Computer Science?
 - Introduction to Python
- 3 Mathematical Software
 - The three big M's
 - Open Source: SAGE
- 4 Summary + Assignments

Purpose of the Course

About the Course

Content

Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's

Open Source: SAGE

Summary + Assignments

Expectations are threefold:

- Understand Basics of Computer Science
- Implement Algorithms in Python
- Use Computer Algebra System Sage

Goals of MCS 260: computer literacy and introduction to programming.

Computer Science is an **active** discipline:
→ learn by doing

Therefore,

- 1 emphasis on five computer projects
- 2 active participation to the lab sessions

Purpose of the Course

About the Course

Content

Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's

Open Source: SAGE

Summary + Assignments

Expectations are threefold:

- Understand Basics of Computer Science
- Implement Algorithms in Python
- Use Computer Algebra System Sage

Goals of MCS 260: computer literacy and introduction to programming.

Computer Science is an **active** discipline:
→ learn by doing

Therefore,

- 1 emphasis on five computer projects
- 2 active participation to the lab sessions

Purpose of the Course

About the Course

Content

Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's

Open Source: SAGE

Summary + Assignments

Expectations are threefold:

- Understand Basics of Computer Science
- Implement Algorithms in Python
- Use Computer Algebra System Sage

Goals of MCS 260: computer literacy and introduction to programming.

Computer Science is an **active** discipline:
→ learn by doing

Therefore,

- 1 emphasis on five computer projects
- 2 active participation to the lab sessions

About the Course

Content
Expectations

Algorithms

What is Computer Science?
Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

Introduction to Computer Science

- 1 About the Course
Content
Expectations
- 2 Algorithms
What is Computer Science?
Introduction to Python
- 3 Mathematical Software
The three big M's
Open Source: SAGE
- 4 Summary + Assignments

What is Computer Science?

Chapter 0 of Computer Science. An Overview

from the Association of Computing Machinery (ACM):

“systematic study of those algorithms which describe and transform information: the underpinning theory, analysis, planning, efficiency, realization and application”

ingredients in definition point at three different subfields:

- 1 theory and analysis: mathematics
- 2 planning, efficiency, and realization: engineering
- 3 applications: business

What is Computer Science?

Chapter 0 of Computer Science. An Overview

from the Association of Computing Machinery (ACM):

“systematic study of those algorithms which describe and transform information: the underpinning theory, analysis, planning, efficiency, realization and application”

ingredients in definition point at three different subfields:

- 1 theory and analysis: mathematics
- 2 planning, efficiency, and realization: engineering
- 3 applications: business

The concept of an algorithm

About the Course

Content
Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

example: compute $\gcd(a, b)$, for $a, b \in \mathbb{N}$, $a \geq b > 0$.

Euclid's algorithm for the greatest common divisor

```
let  $r$  be the remainder of  $a/b$ ;  
if  $r = 0$   
  then  $b$  is the gcd;  
  else the gcd is  $\gcd(b, r)$ ;  
end if.
```

Like a recipe, the main qualities of an algorithm are clarity, correctness, and efficiency.

An algorithm is an ordered set of unambiguous, executable steps that define a terminating process.

The concept of an algorithm

About the Course

Content
Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

example: compute $\gcd(a, b)$, for $a, b \in \mathbb{N}$, $a \geq b > 0$.

Euclid's algorithm for the greatest common divisor

```
let  $r$  be the remainder of  $a/b$ ;  
if  $r = 0$   
  then  $b$  is the gcd;  
  else the gcd is  $\gcd(b, r)$ ;  
end if.
```

Like a recipe, the main qualities of an algorithm are clarity, correctness, and efficiency.

An algorithm is an ordered set of unambiguous, executable steps that define a terminating process.

The concept of an algorithm

About the Course

Content
Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

example: compute $\gcd(a, b)$, for $a, b \in \mathbb{N}$, $a \geq b > 0$.

Euclid's algorithm for the greatest common divisor

```
let  $r$  be the remainder of  $a/b$ ;  
if  $r = 0$   
  then  $b$  is the gcd;  
  else the gcd is  $\gcd(b, r)$ ;  
end if.
```

Like a recipe, the main qualities of an algorithm are clarity, correctness, and efficiency.

An algorithm is an ordered set of unambiguous, executable steps that define a terminating process.

The concept of an algorithm

About the Course

Content
Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

example: compute $\gcd(a, b)$, for $a, b \in \mathbb{N}$, $a \geq b > 0$.

Euclid's algorithm for the greatest common divisor

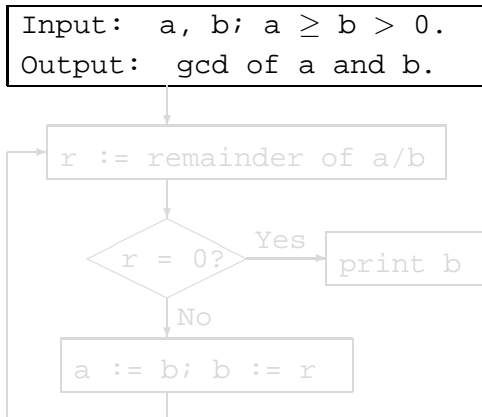
```
let  $r$  be the remainder of  $a/b$ ;  
if  $r = 0$   
  then  $b$  is the gcd;  
  else the gcd is  $\gcd(b, r)$ ;  
end if.
```

Like a recipe, the main qualities of an algorithm are clarity, correctness, and efficiency.

An algorithm is an ordered set of unambiguous, executable steps that define a terminating process.

Flowchart

pictures of algorithms



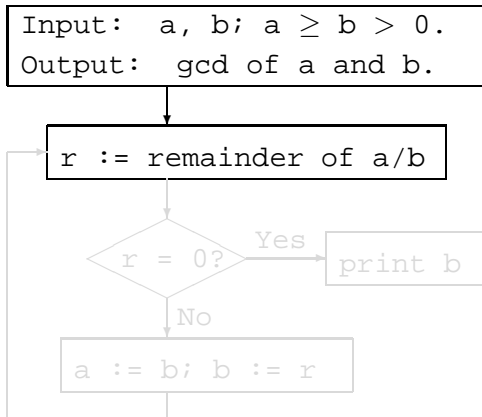
About the
CourseContent
Expectations

Algorithms

What is Computer
Science?Introduction to
PythonMathematical
SoftwareThe three big M's
Open Source: SAGESummary +
Assignments

Flowchart

pictures of algorithms



About the Course

Content
Expectations

Algorithms

What is Computer Science?

Introduction to Python

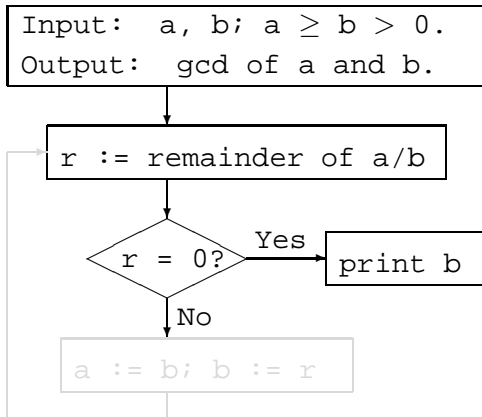
Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

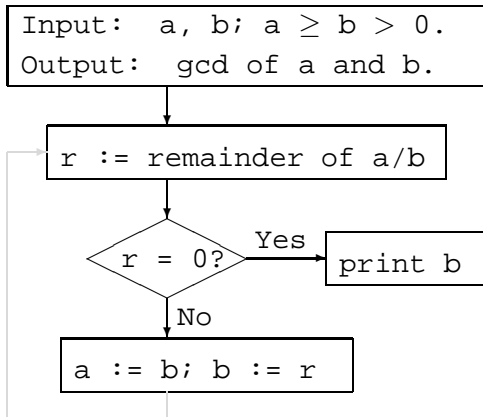
Flowchart

pictures of algorithms



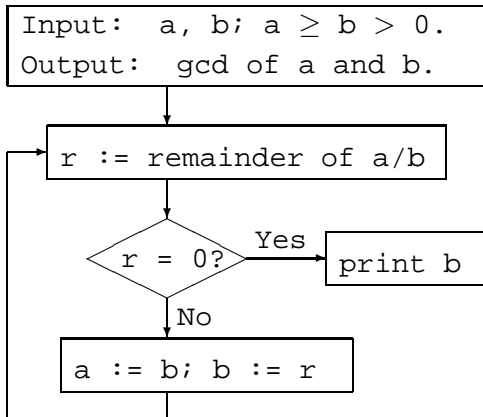
Flowchart

pictures of algorithms



Flowchart

pictures of algorithms



Languages for Programming Algorithms

a quick overview

FORTAN (formula translator) is listed as one of the top ten algorithms of the past century.

COBOL is widely used for business applications.

Algol 60 led to Pascal and Ada, rigorously designed.

C, C++, and Java are perhaps most successful.

Lisp is for functional programming.

Prolog is based on mathematical logic.

Python will be our first programming language.

Languages for Programming Algorithms

a quick overview

FORTRAN (formula translator) is listed as one of the top ten algorithms of the past century.

COBOL is widely used for business applications.

Algol 60 led to Pascal and Ada, rigorously designed.

C, C++, and Java are perhaps most successful.

Lisp is for functional programming.

Prolog is based on mathematical logic.

Python will be our first programming language.

Languages for Programming Algorithms

a quick overview

FORTRAN (formula translator) is listed as one of the top ten algorithms of the past century.

COBOL is widely used for business applications.

Algol 60 led to Pascal and Ada, rigorously designed.

C, C++, and Java are perhaps most successful.

Lisp is for functional programming.

Prolog is based on mathematical logic.

Python will be our first programming language.

Languages for Programming Algorithms

a quick overview

FORTRAN (formula translator) is listed as one of the top ten algorithms of the past century.

COBOL is widely used for business applications.

Algol 60 led to Pascal and Ada, rigorously designed.

C, C++, and Java are perhaps most successful.

Lisp is for functional programming.

Prolog is based on mathematical logic.

Python will be our first programming language.

Languages for Programming Algorithms

a quick overview

FORTAN (formula translator) is listed as one of the top ten algorithms of the past century.

COBOL is widely used for business applications.

Algol 60 led to Pascal and Ada, rigorously designed.

C, C++, and Java are perhaps most successful.

Lisp is for functional programming.

Prolog is based on mathematical logic.

Python will be our first programming language.

Languages for Programming Algorithms

a quick overview

FORTAN (formula translator) is listed as one of the top ten algorithms of the past century.

COBOL is widely used for business applications.

Algol 60 led to Pascal and Ada, rigorously designed.

C, C++, and Java are perhaps most successful.

Lisp is for functional programming.

Prolog is based on mathematical logic.

Python will be our first programming language.

Languages for Programming Algorithms

a quick overview

FORTRAN (formula translator) is listed as one of the top ten algorithms of the past century.

COBOL is widely used for business applications.

Algol 60 led to Pascal and Ada, rigorously designed.

C, C++, and Java are perhaps most successful.

Lisp is for functional programming.

Prolog is based on mathematical logic.

Python will be our first programming language.

About the Course

Content
Expectations

Algorithms

What is Computer Science?

Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

Introduction to Computer Science

- 1 About the Course
Content
Expectations
- 2 Algorithms
What is Computer Science?
Introduction to Python
- 3 Mathematical Software
The three big M's
Open Source: SAGE
- 4 Summary + Assignments

Introduction to Python

Why Python?

programmers prefer rapid application development tools

A Python session:

```
>>> def gcd(a,b):  
...     r = a%b  
...     if r == 0:  
...         return b  
...     else:  
...         return gcd(b,r)  
...  
>>> gcd(12,8)  
4
```

Introduction to Python

Why Python?

About the Course

Content
Expectations

Algorithms

What is Computer Science?
Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

programmers prefer rapid application development tools

A Python session:

```
>>> def gcd(a,b):  
...     r = a%b  
...     if r == 0:  
...         return b  
...     else:  
...         return gcd(b,r)  
...  
>>> gcd(12,8)  
4
```

Overview of Python

history and features

Python is a free, open-source, general-purpose, interpreted, and scripting language for web applications.

History: development started in late 1989
current stable version is 2.6.4 (2.5 ok too)
Python 3.0 (and up) is a revision, not
guaranteed to be backwards compatible

Features: easy, scalable, high level, object oriented,
interpreted, extensible and flexible, rich core
library, manages memory, support web
applications, object distribution, databases,
GUI programming, extendable and
embeddable, exception handling, portable,
freeware.

Visit www.python.org for documentation and resources.

About the Course

Content
Expectations

Algorithms

What is Computer
Science?

Introduction to
Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

Overview of Python

history and features

Python is a free, open-source, general-purpose, interpreted, and scripting language for web applications.

History: development started in late 1989
current stable version is 2.6.4 (2.5 ok too)
Python 3.0 (and up) is a revision, not
guaranteed to be backwards compatible

Features: easy, scalable, high level, object oriented, interpreted, extensible and flexible, rich core library, manages memory, support web applications, object distribution, databases, GUI programming, extendable and embeddable, exception handling, portable, freeware.

Visit www.python.org for documentation and resources.

Overview of Python

history and features

Python is a free, open-source, general-purpose, interpreted, and scripting language for web applications.

History: development started in late 1989
current stable version is 2.6.4 (2.5 ok too)
Python 3.0 (and up) is a revision, not
guaranteed to be backwards compatible

Features: easy, scalable, high level, object oriented,
interpreted, extensible and flexible, rich core
library, manages memory, support web
applications, object distribution, databases,
GUI programming, extendable and
embeddable, exception handling, portable,
freeware.

Visit www.python.org for documentation and resources.

Overview of Python

history and features

Python is a free, open-source, general-purpose, interpreted, and scripting language for web applications.

History: development started in late 1989
current stable version is 2.6.4 (2.5 ok too)
Python 3.0 (and up) is a revision, not
guaranteed to be backwards compatible

Features: easy, scalable, high level, object oriented,
interpreted, extensible and flexible, rich core
library, manages memory, support web
applications, object distribution, databases,
GUI programming, extendable and
embeddable, exception handling, portable,
freeware.

Visit www.python.org for documentation and resources.

Overview of Python

history and features

Python is a free, open-source, general-purpose, interpreted, and scripting language for web applications.

History: development started in late 1989
current stable version is 2.6.4 (2.5 ok too)
Python 3.0 (and up) is a revision, not
guaranteed to be backwards compatible

Features: easy, scalable, high level, object oriented,
interpreted, extensible and flexible, rich core
library, manages memory, support web
applications, object distribution, databases,
GUI programming, extendable and
embeddable, exception handling, portable,
freeware.

Visit www.python.org for documentation and resources.

Overview of Python

history and features

Python is a free, open-source, general-purpose, interpreted, and scripting language for web applications.

History: development started in late 1989
current stable version is 2.6.4 (2.5 ok too)
Python 3.0 (and up) is a revision, not
guaranteed to be backwards compatible

Features: easy, scalable, high level, object oriented, interpreted, extensible and flexible, rich core library, manages memory, support web applications, object distribution, databases, GUI programming, extendable and embeddable, exception handling, portable, freeware.

Visit www.python.org for documentation and resources.

About the
CourseContent
Expectations

Algorithms

What is Computer
Science?Introduction to
PythonMathematical
SoftwareThe three big M's
Open Source: SAGESummary +
Assignments

Application Areas & Comparisons

Python is mainly used

- to create prototypes of an application
- to glue together large software applications
- to write CGI scripts for the internet

Compared to other languages, Python

- is slower than compiled languages
- is good tool to test C and C++ programs
- has nice syntax, nicer than other languages

JPython is Python interpreter, written in Java
given Python code, Cython generates C code

Application Areas & Comparisons

Python is mainly used

- to create prototypes of an application
- to glue together large software applications
- to write CGI scripts for the internet

Compared to other languages, Python

- is slower than compiled languages
- is good tool to test C and C++ programs
- has nice syntax, nicer than other languages

JPython is Python interpreter, written in Java
given Python code, Cython generates C code

Application Areas & Comparisons

Python is mainly used

- to create prototypes of an application
- to glue together large software applications
- to write CGI scripts for the internet

Compared to other languages, Python

- is slower than compiled languages
- is good tool to test C and C++ programs
- has nice syntax, nicer than other languages

JPython is Python interpreter, written in Java
given Python code, Cython generates C code

About the Course

Content
Expectations

Algorithms

What is Computer Science?
Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

Introduction to Computer Science

- 1 About the Course
Content
Expectations
- 2 Algorithms
What is Computer Science?
Introduction to Python
- 3 Mathematical Software
The three big M's
Open Source: SAGE
- 4 Summary + Assignments

The three big M's

commercial mathematical software

All labs on campus have the following software

Maple: a computer algebra software system, computer algebra implements algorithms in symbolic and numerical computation.

Mathematica: another major computer algebra system with a notebook interface and a logical programming language.

Matlab: a system for scientific computing, its main original strength is numerical linear algebra.

All these systems implement a wide variety of algorithms and provide support for high level data structures.

These systems allow to test new algorithms very quickly as their programming capabilities are easy to use.

The three big M's

commercial mathematical software

All labs on campus have the following software

Maple: a computer algebra software system, computer algebra implements algorithms in symbolic and numerical computation.

Mathematica: another major computer algebra system with a notebook interface and a logical programming language.

Matlab: a system for scientific computing, its main original strength is numerical linear algebra.

All these systems implement a wide variety of algorithms and provide support for high level data structures.

These systems allow to test new algorithms very quickly as their programming capabilities are easy to use.

The three big M's

commercial mathematical software

All labs on campus have the following software

Maple: a computer algebra software system, computer algebra implements algorithms in symbolic and numerical computation.

Mathematica: another major computer algebra system with a notebook interface and a logical programming language.

Matlab: a system for scientific computing, its main original strength is numerical linear algebra.

All these systems implement a wide variety of algorithms and provide support for high level data structures.

These systems allow to test new algorithms very quickly as their programming capabilities are easy to use.

The three big M's

commercial mathematical software

All labs on campus have the following software

Maple: a computer algebra software system, computer algebra implements algorithms in symbolic and numerical computation.

Mathematica: another major computer algebra system with a notebook interface and a logical programming language.

Matlab: a system for scientific computing, its main original strength is numerical linear algebra.

All these systems implement a wide variety of algorithms and provide support for high level data structures.

These systems allow to test new algorithms very quickly as their programming capabilities are easy to use.

The three big M's

commercial mathematical software

All labs on campus have the following software

Maple: a computer algebra software system, computer algebra implements algorithms in symbolic and numerical computation.

Mathematica: another major computer algebra system with a notebook interface and a logical programming language.

Matlab: a system for scientific computing, its main original strength is numerical linear algebra.

All these systems implement a wide variety of algorithms and provide support for high level data structures.

These systems allow to test new algorithms very quickly as their programming capabilities are easy to use.

The three big M's

commercial mathematical software

All labs on campus have the following software

Maple: a computer algebra software system, computer algebra implements algorithms in symbolic and numerical computation.

Mathematica: another major computer algebra system with a notebook interface and a logical programming language.

Matlab: a system for scientific computing, its main original strength is numerical linear algebra.

All these systems implement a wide variety of algorithms and provide support for high level data structures.

These systems allow to test new algorithms very quickly as their programming capabilities are easy to use.

About the Course

Content
Expectations

Algorithms

What is Computer Science?
Introduction to Python

Mathematical Software

The three big M's
Open Source: SAGE

Summary + Assignments

Introduction to Computer Science

- 1 About the Course
Content
Expectations
- 2 Algorithms
What is Computer Science?
Introduction to Python
- 3 Mathematical Software
The three big M's
Open Source: SAGE
- 4 Summary + Assignments

Open Source: SAGE

Software for Algebra and Geometry
Experimentation

SAGE is a computer algebra system written in Python and a modified version of Pyrex. www.sagemath.org

- SAGE can be used to study a huge range of mathematics.
- SAGE includes interfaces to Magma, Maple, Mathematica, MATLAB, and MuPAD, and the free programs Axiom, GAP, GP/PARI, Macaulay2, Maxima, Octave, and Singular.
- SAGE can be used from your web browser either connecting to programs running on your computer or running somewhere else.

SAGE is available as free software under the GNU GPL

Open Source: SAGE

Software for Algebra and Geometry
Experimentation

SAGE is a computer algebra system written in Python and a modified version of Pyrex. www.sagemath.org

- SAGE can be used to study a huge range of mathematics.
- SAGE includes interfaces to Magma, Maple, Mathematica, MATLAB, and MuPAD, and the free programs Axiom, GAP, GP/PARI, Macaulay2, Maxima, Octave, and Singular.
- SAGE can be used from your web browser either connecting to programs running on your computer or running somewhere else.

SAGE is available as free software under the GNU GPL

Open Source: SAGE

Software for Algebra and Geometry
Experimentation

SAGE is a computer algebra system written in Python and a modified version of Pyrex. www.sagemath.org

- SAGE can be used to study a huge range of mathematics.
- SAGE includes interfaces to Magma, Maple, Mathematica, MATLAB, and MuPAD, and the free programs Axiom, GAP, GP/PARI, Macaulay2, Maxima, Octave, and Singular.
- SAGE can be used from your web browser either connecting to programs running on your computer or running somewhere else.

SAGE is available as free software under the GNU GPL

Open Source: SAGE

Software for Algebra and Geometry
Experimentation

SAGE is a computer algebra system written in Python and a modified version of Pyrex. www.sagemath.org

- SAGE can be used to study a huge range of mathematics.
- SAGE includes interfaces to Magma, Maple, Mathematica, MATLAB, and MuPAD, and the free programs Axiom, GAP, GP/PARI, Macaulay2, Maxima, Octave, and Singular.
- SAGE can be used from your web browser either connecting to programs running on your computer or running somewhere else.

SAGE is available as free software under the GNU GPL

Open Source: SAGE

Software for Algebra and Geometry
Experimentation

SAGE is a computer algebra system written in Python and a modified version of Pyrex. www.sagemath.org

- SAGE can be used to study a huge range of mathematics.
- SAGE includes interfaces to Magma, Maple, Mathematica, MATLAB, and MuPAD, and the free programs Axiom, GAP, GP/PARI, Macaulay2, Maxima, Octave, and Singular.
- SAGE can be used from your web browser either connecting to programs running on your computer or running somewhere else.

SAGE is available as free software under the GNU GPL

Summary + Assignments

In this lecture we covered

- *Computer Science*: chapter 0 and section 5.1: the concept of an algorithm;
- the introduction of *Python Programming*.

On Tuesday or Thursday, go to lab SES 205B

- make sure your netid is working,
- there will be a quiz at the end.

Assignments:

- 1 Explain the differences between formulas and algorithms. What are the similarities?
- 2 Write down the instructions to walk from Lincoln Hall to SES 205.
- 3 Describe the algorithm to compute a/b , with a and b given as sequences of digits.