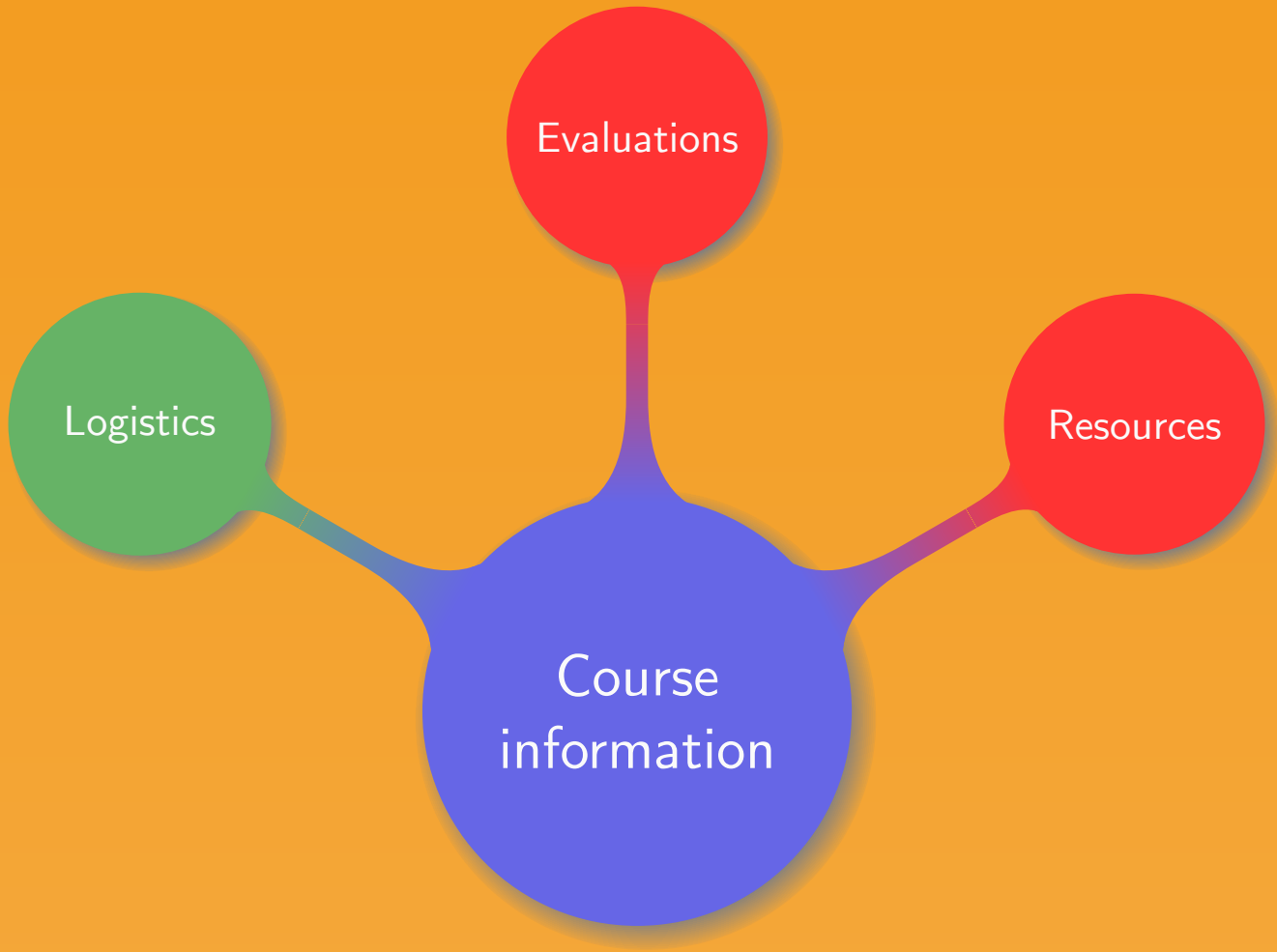
Introduction to Computer and Programming

0. Course information

Manuel – Summer 2019

Chapter organisation



Basic information

Teaching team:

- Instructor: Manuel (charlem@sjtu.edu.cn)
- Teaching assistants:
  - Yifei (zhangyifei-chelsea@sjtu.edu.cn)
  - Xiwen (victoria-x@sjtu.edu.cn)
  - Zekai (sleepingring@sjtu.edu.cn)
  - Zhengyuan (zhangzhengyuan@sjtu.edu.cn)

Basic information

Teaching team:

- Instructor: Manuel (charlem@sjtu.edu.cn)
- Teaching assistants:
  - Yifei (zhangyifei-chelsea@sjtu.edu.cn)
  - Xiwen (victoria-x@sjtu.edu.cn)
  - Zekai (sleepingring@sjtu.edu.cn)
  - Zhengyuan (zhangzhengyuan@sjtu.edu.cn)

Important rules:

- When contacting a TA for an important matter, `CC` the instructor
- Prepend [VG101] to the subject, e.g. `Subject: [VG101] Grades`
- Use SJTU jBox service to share large files ($>$ 2 MB)

Basic information

Teaching team:

- Instructor: Manuel (charlem@sjtu.edu.cn)
- Teaching assistants:
  - Yifei (zhangyifei-chelsea@sjtu.edu.cn)
  - Xiwen (victoria-x@sjtu.edu.cn)
  - Zekai (sleepingring@sjtu.edu.cn)
  - Zhengyuan (zhangzhengyuan@sjtu.edu.cn)

Important rules:

- When contacting a TA for an important matter, `CC` the instructor
- Prepend [VG101] to the subject, e.g. `Subject: [VG101] Grades`
- Use SJTU jBox service to share large files ($>$ 2 MB)

Never send large files by email

Course schedule

Course arrangements:

- Lectures:
  - Tuesday 14:00 – 15:40
  - Thursday 14:00 – 15:40
  - Friday 14:00 – 15:40 (weeks 1-4, 12)
- Office hours: Tuesday 15:40 – 17:50

Appointments outside of the office hours can be taken by email

Course objectives

Primary goals:

- Understand the main concepts of computer and programming
- Design simple algorithms
- Implement clearly stated algorithms in MATLAB, C, and C++

Course objectives

Primary goals:

- Understand the main concepts of computer and programming
- Design simple algorithms
- Implement clearly stated algorithms in MATLAB, C, and C++

Be able to quickly adjust to new languages and libraries

Course workflow

Learning strategy:

- Course side:
  1. Understand the basics on computers
  2. Get familiar with programming through MATLAB
  3. Understand deeper concepts with C
  4. Bridge the gap between computers and humans using C++

Course workflow

Learning strategy:

- Course side:
  1. Understand the basics on computers
  2. Get familiar with programming through MATLAB
  3. Understand deeper concepts with C
  4. Bridge the gap between computers and humans using C++
- Personal side:
  1. Read and write code

Course workflow

Learning strategy:

- Course side:
  1. Understand the basics on computers
  2. Get familiar with programming through MATLAB
  3. Understand deeper concepts with C
  4. Bridge the gap between computers and humans using C++
- Personal side:
  1. Read and write code
  2. Write more code

Course workflow

Learning strategy:

- Course side:
  1. Understand the basics on computers
  2. Get familiar with programming through MATLAB
  3. Understand deeper concepts with C
  4. Bridge the gap between computers and humans using C++
- Personal side:
  1. Read and write code
  2. Write more code
  3. Write even more code

Course workflow

Learning strategy:

- Course side:
  1. Understand the basics on computers
  2. Get familiar with programming through MATLAB
  3. Understand deeper concepts with C
  4. Bridge the gap between computers and humans using C++
- Personal side:
  1. Read and write code
  2. Write more code
  3. Write even more code
  4. Do not stop writing code

Course workflow

Learning strategy:

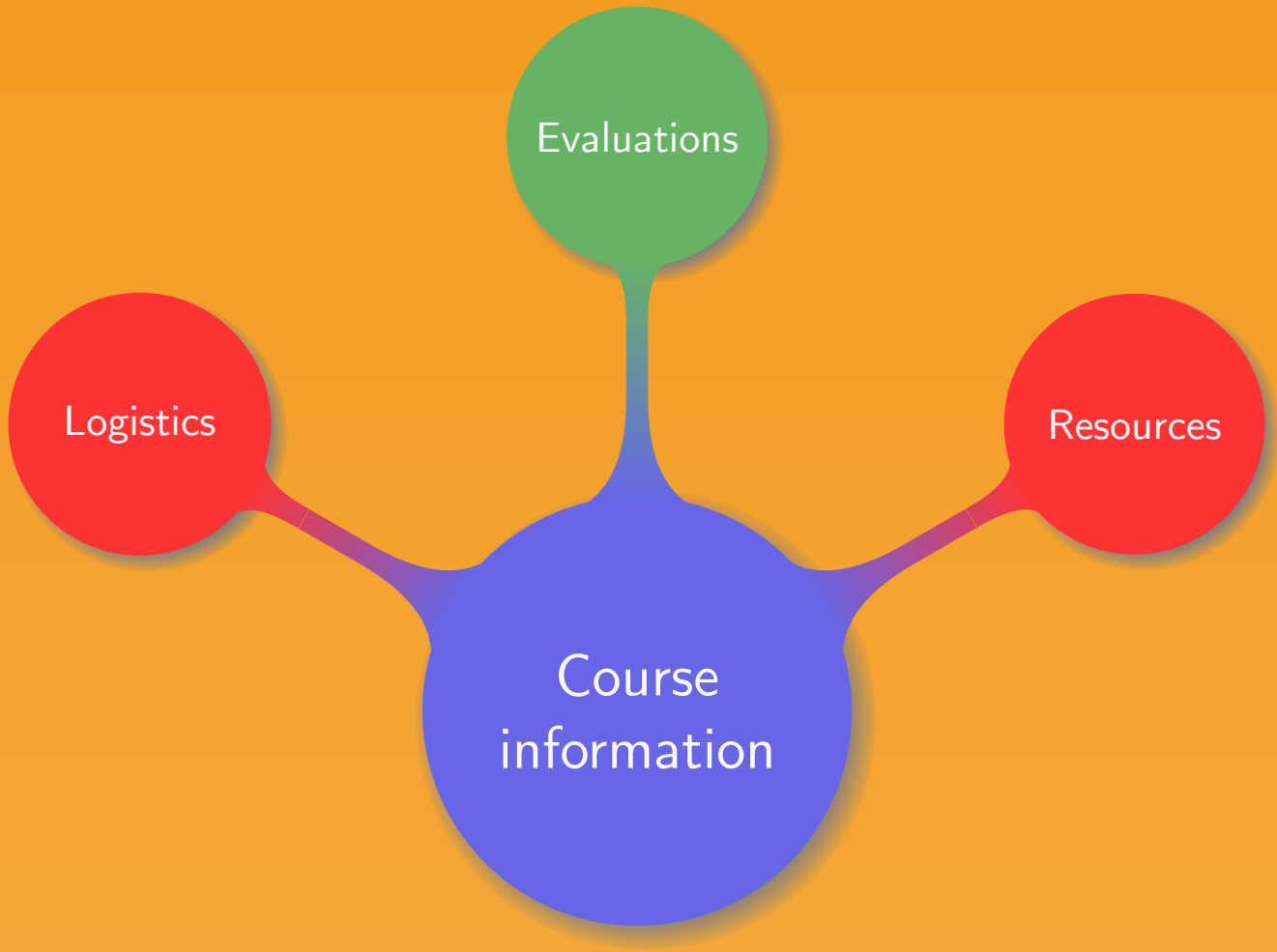
- Course side:
  1. Understand the basics on computers
  2. Get familiar with programming through MATLAB
  3. Understand deeper concepts with C
  4. Bridge the gap between computers and humans using C++
- Personal side:
  1. Read and write code
  2. Write more code
  3. Write even more code
  4. Do not stop writing code
  5. Relate known strategies to new problems
  6. Perform extra research

Course outcomes

Detailed goals:

- Proficiency with data representation and naming
- Proficiency with data input and output
- Proficiency with programming with math and logical operators and functions
- Proficiency with designing, testing, and implementing functions and procedures
- Proficiency with control flow using selection and iteration
- Proficiency with use of pre-defined data structures
- Proficiency with primitive and complex data types
- Proficiency with visualization of data
- Proficiency with algorithm design for engineering analysis

Chapter organisation



Assignments

Homework:

- Total: 8
- Content: basic algorithms, Matlab, C, and C++

Labs:

- Total: 8
- Content: guided sessions in Matlab, C, and C++

Projects:

- Total: 3
- Content: advanced problems in Matlab, C, and C++

Challenges:

- Total: 1
- Content: write a Gomoku AI

Grading policy

Grade weighting:

- Matlab midterm: 20%
- C midterm: 20%
- C++ final: 20%
- Projects: 35%
- Labs: 5%

Grading policy

Grade weighting:

- Matlab midterm: 20%
- C midterm: 20%
- C++ final: 20%
- Projects: 35%
- Labs: 5%

Assignment submissions: -10% per day, not accepted after 3 days

Grades will be curved to balance the three sections

Grading policy

Homework:

- Not graded, completed in groups
- Each student must complete all the mandatory exercises
- Each student must review the code of at least one teammate
- A final improved version must be submitted for each group
- Submissions should be successfully compiled or interpreted
- Group discussions must take place on Piazza

Grading policy

Homework:

- Not graded, completed in groups
- Each student must complete all the mandatory exercises
- Each student must review the code of at least one teammate
- A final improved version must be submitted for each group
- Submissions should be successfully compiled or interpreted
- Group discussions must take place on Piazza

Students not following guidelines will receive large deductions on their final course grade

Honor Code

General rules:

- Not allowed:
  - Reuse the code or work from other students or groups
  - Reuse the code or work from the internet
  - Share too many details on how to complete a task

Honor Code

General rules:

- Not allowed:
  - Reuse the code or work from other students or groups
  - Reuse the code or work from the internet
  - Share too many details on how to complete a task
- Allowed:
  - Reuse part the course or textbooks and quoting the source
  - Share ideas and understandings on the course
  - Provide hints on where or how to find information

Honor Code

Documents allowed during the exams:

- Part A: a mono or bilingual dictionary
- Part B:
  - The lecture slides with **notes on them** (paper or electronic)
  - A mono or bilingual dictionary

Group works:

- Every student in a group is responsible for his group's submission
- If a student breaks the Honor Code, the whole group is guilty

Special circumstances

Contact us as early as possible when:

- Facing special circumstances, e.g. full time work, illness, etc.
- Feeling late in the course
- Feeling to work hard without any result

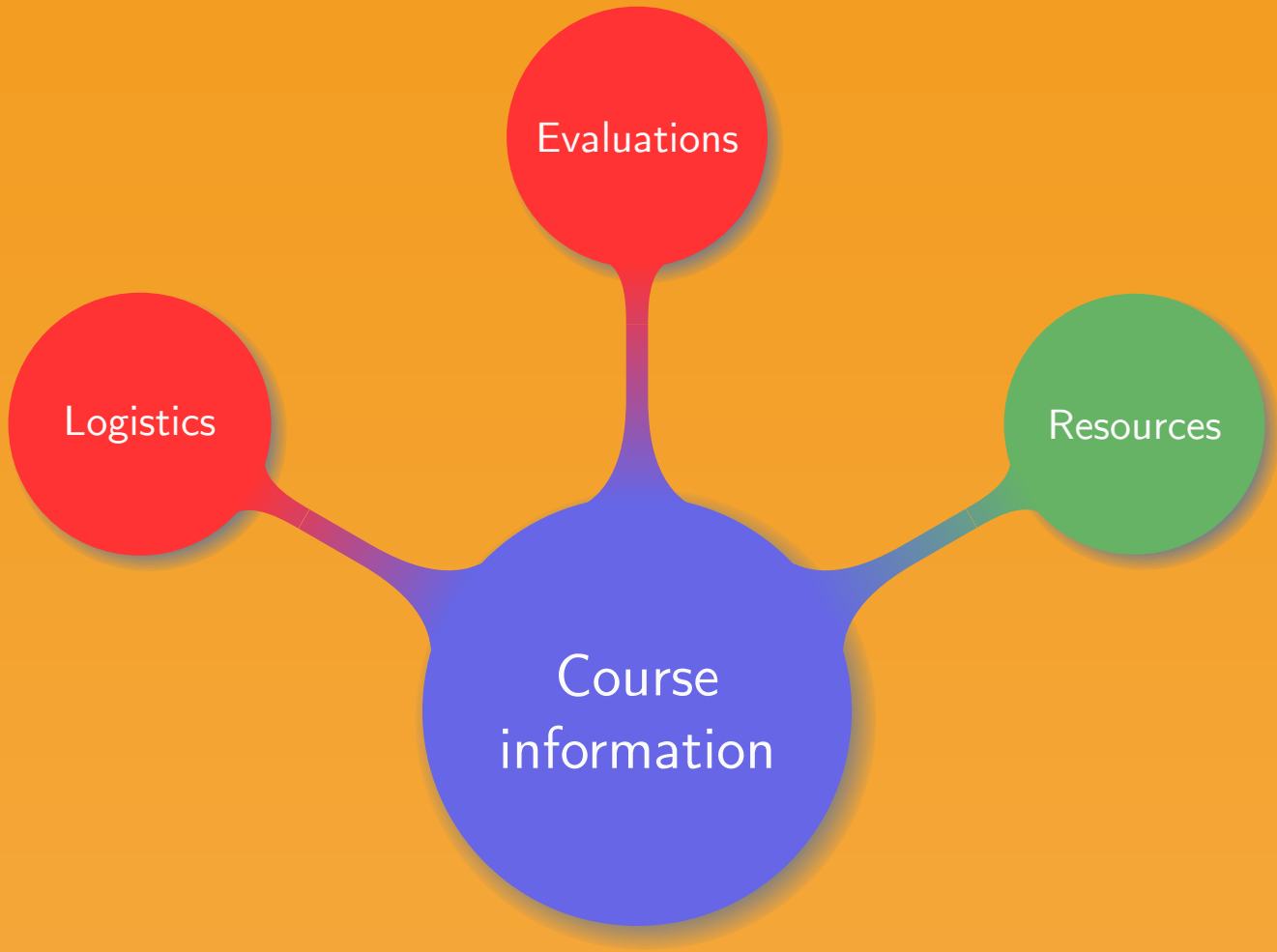
Special circumstances

Contact us as early as possible when:

- Facing special circumstances, e.g. full time work, illness, etc.
- Feeling late in the course
- Feeling to work hard without any result

Any late request will be rejected

Chapter organisation



Canvas

Information and documents available on the Canvas platform:

- Course materials:
  - Syllabus
  - Lecture slides
  - Homework
  - Labs
  - Projects
  - Challenges
- Course information:
  - Announcements
  - Notifications
  - Grades
  - Polls

References

Useful places where to find information:

- MATLAB documentation
- *C for Engineers and Scientists* by Harry H. Cheng
- *Thinking in C++* by Bruce Eckel
- Search information online, i.e. $\{websites \setminus \{local\ Chinese\ network\}\}$

References

Useful places where to find information:

- MATLAB documentation
- *C for Engineers and Scientists* by Harry H. Cheng
- *Thinking in C++* by Bruce Eckel
- Search information online, i.e. $\{websites \setminus \{local\ Chinese\ network\}\}$

Never use Baidu in any course

Key points

- Work regularly, do not wait the last minute/day
- Respect the Honor Code
- Go beyond what is taught
- Do not learn, understand
- Keep in touch with us
- Advice and suggestions are always much appreciated

Thank you!