Algorithms and Complexity

0. Organisational Matters

Marc Hellmuth

University of Stockholm

Algorithms and Complexity

Lecturer: Marc Hellmuth (marc.hellmuth@math.su.se)

TAs: Anna Lindeberg (anna.lindeberg@math.su.se)
Axel Ljungström (axel.ljungstrom@math.su.se)

Algorithms and Complexity

Lecturer: Marc Hellmuth (marc.hellmuth@math.su.se)

TAs: Anna Lindeberg (anna.lindeberg@math.su.se)
Axel Ljungström (axel.ljungstrom@math.su.se)

Lectures are weekly: MO and THU 1 - 3pm
Tutorials are weekly: MO and THU 3 - 4pm

Algorithms and Complexity

Lecturer: Marc Hellmuth (marc.hellmuth@math.su.se)

TAs: Anna Lindeberg (anna.lindeberg@math.su.se)
Axel Ljungström (axel.ljungstrom@math.su.se)

Lectures are weekly: MO and THU 1 - 3pm
Tutorials are weekly: MO and THU 3 - 4pm

However, as I am on travel on two weeks, the schedule will be slightly adjusted.
First changes:

additional lecture: Monday 02-sep 3-4
additional lecture: Monday 16-sep 3-4

there will be additional adjustments (TBA on homepage - so check frequently)

Algorithms and Complexity

Lecturer: Marc Hellmuth (marc.hellmuth@math.su.se)

TAs: Anna Lindeberg (anna.lindeberg@math.su.se)
Axel Ljungström (axel.ljungstrom@math.su.se)

Lectures are weekly: MO and THU 1 - 3pm
Tutorials are weekly: MO and THU 3 - 4pm

However, as I am on travel on two weeks, the schedule will be slightly adjusted.
First changes:

additional lecture: Monday 02-sep 3-4
additional lecture: Monday 16-sep 3-4

there will be additional adjustments (TBA on homepage - so check frequently)

All lectures and tutorial take place in Lecture room 1 (except in week 43)

See SCHEDULE-LINK at kurser homepage.

Algorithms and Complexity

Lecturer: Marc Hellmuth (marc.hellmuth@math.su.se)

TAs: Anna Lindeberg (anna.lindeberg@math.su.se)
Axel Ljungström (axel.ljungstrom@math.su.se)

Lectures are weekly: MO and THU 1 - 3pm
Tutorials are weekly: MO and THU 3 - 4pm

However, as I am on travel on two weeks, the schedule will be slightly adjusted.
First changes:

additional lecture: Monday 02-sep 3-4
additional lecture: Monday 16-sep 3-4

there will be additional adjustments (TBA on homepage - so check frequently)

All lectures and tutorial take place in Lecture room 1 (except in week 43)

See SCHEDULE-LINK at kurser homepage.

All information / news / exercises etc.pp. can be found online:
https://kurser.math.su.se/ Course: DA4005 HT24

Algorithms and Complexity

Lectures will be in-person.

Recordings of former lectures, and the current slides and a (handwritten) script are available online (see DA4005 homepage).

Course examination

Course examination is done in four parts:

- Home assignments IND1 and IND2, worth 1.5 HP each, graded A-F.
- Practical exercises (PE), worth 1.5 HP, graded P/F.
- A written exam (THEO), worth 3 HP and graded A-F.

Course examination

Course examination is done in four parts:

- Home assignments IND1 and IND2, worth 1.5 HP each, graded A-F.
- Practical exercises (PE), worth 1.5 HP, graded P/F.
- A written exam (THEO), worth 3 HP and graded A-F.

- INDi consists of 2 individual exercise sheets, $i \in \{1, 2\}$
- PE consists of 2 individual programming projects

Course examination

Course examination is done in four parts:

- Home assignments IND1 and IND2, worth 1.5 HP each, graded A-F.
- Practical exercises (PE), worth 1.5 HP, graded P/F.
- A written exam (THEO), worth 3 HP and graded A-F.

- INDi consists of 2 individual exercise sheets, $i \in \{1, 2\}$
- PE consists of 2 individual programming projects

You pass the course if IND1, IND2, PE and exam have been passed.

Course examination

Course examination is done in four parts:

- Home assignments IND1 and IND2, worth 1.5 HP each, graded A-F.
- Practical exercises (PE), worth 1.5 HP, graded P/F.
- A written exam (THEO), worth 3 HP and graded A-F.

- INDi consists of 2 individual exercise sheets, $i \in \{1, 2\}$
- PE consists of 2 individual programming projects

You pass the course if IND1, IND2, PE and exam have been passed.

To pass INDi at least 50% of the exercises must be correct (in total) $i \in \{1, 2\}$.
To pass PE each individual programming exercise must be passed.
To pass THEO, at least 50% of THEO must be correct.

Course examination

Course examination is done in four parts:

- Home assignments IND1 and IND2, worth 1.5 HP each, graded A-F.
- Practical exercises (PE), worth 1.5 HP, graded P/F.
- A written exam (THEO), worth 3 HP and graded A-F.

- INDi consists of 2 individual exercise sheets, $i \in \{1, 2\}$
- PE consists of 2 individual programming projects

You pass the course if IND1, IND2, PE and exam have been passed.

To pass INDi at least 50% of the exercises must be correct (in total) $i \in \{1, 2\}$.
To pass PE each individual programming exercise must be passed.
To pass THEO, at least 50% of THEO must be correct.

All solutions must be provided in English!

Course examination

Course examination is done in four parts:

- Home assignments IND1 and IND2, worth 1.5 HP each, graded A-F.
- Practical exercises (PE), worth 1.5 HP, graded P/F.
- A written exam (THEO), worth 3 HP and graded A-F.

- INDi consists of 2 individual exercise sheets, $i \in \{1, 2\}$
- PE consists of 2 individual programming projects

Team work to discuss the exercises is allowed and also recommended.
BUT:

- everyone has to hand in an individual and independent solution of the exercies
- you must be able to explain your solutions in the tutorial
- no copies of solutions
- never forget name + student number

Course examination

There are deadlines when to hand-in specified at the DA4005 webpage and ex-sheets.

Course examination

There are deadlines when to hand-in specified at the DA4005 webpage and ex-sheets.

Hand in the respectice exercise before the end of the deadline as follows:

Course examination

There are deadlines when to hand-in specified at the DA4005 webpage and ex-sheets.

Hand in the respectice exercise before the end of the deadline as follows:

- IND1- and IND2-exercises should be provided as single PDF-file (handwritten-scanned ot latex-PDF).

Course examination

There are deadlines when to hand-in specified at the DA4005 webpage and ex-sheets.

Hand in the respectice exercise before the end of the deadline as follows:

- IND1- and IND2-exercises should be provided as single PDF-file (handwritten-scanned ot latex-PDF).
- PE-exercises consists of programming-tasks and should be provided as single compressed zip-file in case you have more than one file.

Course examination

There are deadlines when to hand-in specified at the DA4005 webpage and ex-sheets.

Hand in the respectice exercise before the end of the deadline as follows:

- IND1- and IND2-exercises should be provided as single PDF-file (handwritten-scanned ot latex-PDF).
- PE-exercises consists of programming-tasks and should be provided as single compressed zip-file in case you have more than one file.

  When handing in programming exercises, always document how to compile and run your program.

  Well-commented source code is required! Do not copy source-code from WWW!

Course examination

There are deadlines when to hand-in specified at the DA4005 webpage and ex-sheets.

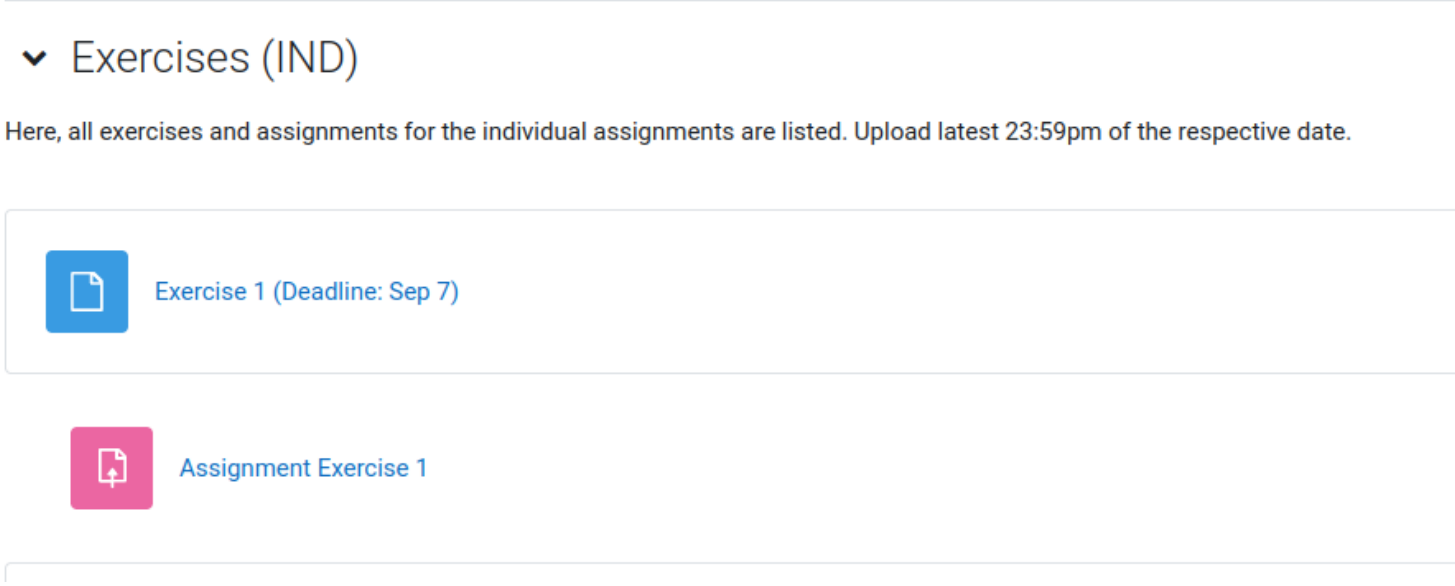
Hand in the respectice exercise before the end of the deadline as follows:

- IND1- and IND2-exercises should be provided as single PDF-file (handwritten-scanned ot latex-PDF).
- PE-exercises consists of programming-tasks and should be provided as single compressed zip-file in case you have more than one file.

  When handing in programming exercises, always document how to compile and run your program.

  Well-commented source code is required! Do not copy source-code from WWW!
- On the IND/PE- exercises also provide your individual student email `abcd1234@student.su.se`
- upload the files at the course homepage under the respective assignment link.

Course examination



TIME MANAGEMENT

TIME MANAGEMENT

SU homepage: *"1.5 HEC =* 40 *hours"*

DA 4005 = 7.5 HP = *total* $200h$

TIME MANAGEMENT

SU homepage: *"1.5 HEC = 40 hours"*

DA 4005 = 7.5 HP = *total* $200h$

- IND1 (2 Exercises) = 1.5 HP = *total* $40h$
- IND2 (2 Exercises) = 1.5 HP = *total* $40h$
- PE (2 Exercises) = 1.5 HP = *total* $40h$
- A written exam ("THEO") = 3 HP = *total* $80h$

Hence each Exercise corresponds to $\sim 20h$.

The exercises are *not* super difficult but possibly time-intensive!

START EARLY!

TIME MANAGEMENT

SU homepage: *"1.5 HEC = 40 hours"*

DA 4005 = 7.5 HP = *total* $200h$

- IND1 (2 Exercises) = 1.5 HP = *total* $40h$
- IND2 (2 Exercises) = 1.5 HP = *total* $40h$
- PE (2 Exercises) = 1.5 HP = *total* $40h$
- A written exam ("THEO") = 3 HP = *total* $80h$

Hence each Exercise corresponds to $\sim 20h$.

The exercises are *not* super difficult but possibly time-intensive!

START EARLY!
START EARLY!

TIME MANAGEMENT

SU homepage: *"1.5 HEC = 40 hours"*

DA 4005 = 7.5 HP = *total* $200h$

- IND1 (2 Exercises) = 1.5 HP = *total* $40h$
- IND2 (2 Exercises) = 1.5 HP = *total* $40h$
- PE (2 Exercises) = 1.5 HP = *total* $40h$
- A written exam ("THEO") = 3 HP = *total* $80h$

Hence each Exercise corresponds to $\sim 20h$.

The exercises are *not* super difficult but possibly time-intensive!

START EARLY!
START EARLY!
START EARLY!

About this course

I expect that all students are familiar with big-O, big-Ω, ... notation and are able to determine the runtime of a given algorithm!

If this does not apply for you watch, the additional video ”CrashCourseRuntime” in folder ”Part 2 Complexity”!

About this course

I expect that all students are familiar with big-O, big-Ω, ... notation and are able to determine the runtime of a given algorithm!

If this does not apply for you watch, the additional video ”CrashCourseRuntime” in folder ”Part 2 Complexity”!

Content:

Why?

- algorithms are central in computer science
- Design of algorithms:
  - Is algorithm correct?
  - Does algorithm terminate?
  - Is algorithm efficient?
- Efficiency: depends often on the used datastructure.

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency
- compare alternative algorithms and data structures with respect to efficiency and reliability

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency
- compare alternative algorithms and data structures with respect to efficiency and reliability
- define the concepts P, NP, NP-completeness and undecidability

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency
- compare alternative algorithms and data structures with respect to efficiency and reliability
- define the concepts P, NP, NP-completeness and undecidability
- compare problems with respect to complexity using reductions,

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency
- compare alternative algorithms and data structures with respect to efficiency and reliability
- define the concepts P, NP, NP-completeness and undecidability
- compare problems with respect to complexity using reductions,
- explain how problems of high complexity can be handled

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency
- compare alternative algorithms and data structures with respect to efficiency and reliability
- define the concepts P, NP, NP-completeness and undecidability
- compare problems with respect to complexity using reductions,
- explain how problems of high complexity can be handled
- independently construct computer programs that use time and memory efficiently

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency
- compare alternative algorithms and data structures with respect to efficiency and reliability
- define the concepts P, NP, NP-completeness and undecidability
- compare problems with respect to complexity using reductions,
- explain how problems of high complexity can be handled
- independently construct computer programs that use time and memory efficiently
- in professional life identify and attack problems that are unrealistically resource demanding or not possible to solve on a computer

Aim

After taking the course the student shall be able to:

- develop and implement algorithms with data structures and analyze them with respect to correctness and efficiency
- compare alternative algorithms and data structures with respect to efficiency and reliability
- define the concepts P, NP, NP-completeness and undecidability
- compare problems with respect to complexity using reductions,
- explain how problems of high complexity can be handled
- independently construct computer programs that use time and memory efficiently
- in professional life identify and attack problems that are unrealistically resource demanding or not possible to solve on a computer

Let’s Get It Started !