

Introduction & overview

□

Jeremy Yallop

`jeremy.yallop@cl.cam.ac.uk`

About the class

Intraseminar structure

Structure



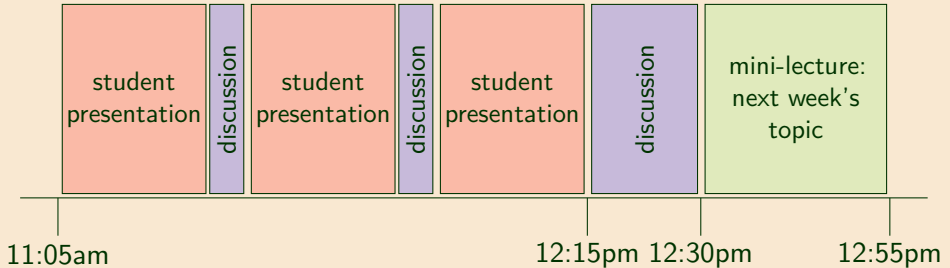
High marks

Low marks

PL

Languages

Processing



Interseminar structure

Structure



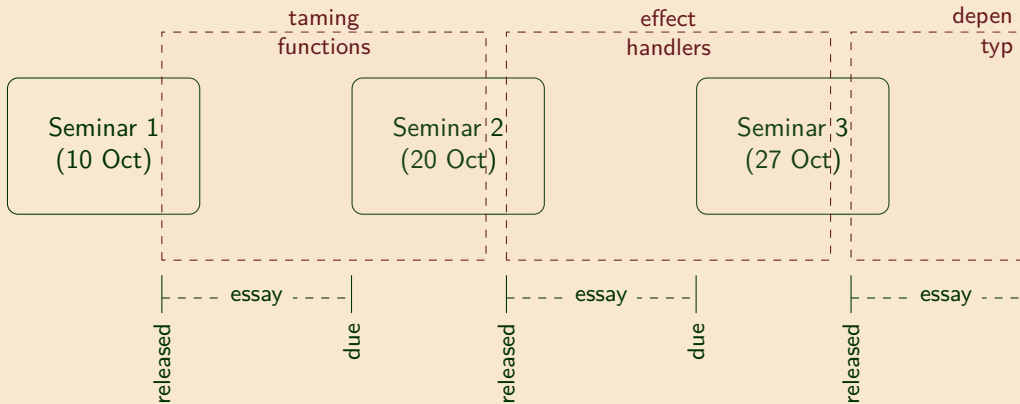
High marks

Low marks

PL

Languages

Processing

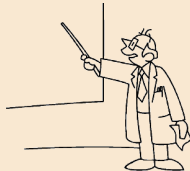


What you'll do each week

Structure



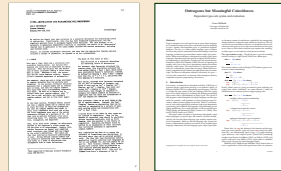
Introductory mini-lecture



Background reading



Read papers



Optional: wider reading



Write & submit essay



Take part in discussion



High marks

Low marks

PL

Languages

Processing

Presentation slot assignments

Structure



High marks

Low marks

PL

Languages

Processing

Date	Topic	Speaker 1	Speaker 2	Speaker 3
20 Oct	Taming functions	TBD	TBD	TBD
27 Oct	Effect handlers	TBD	TBD	TBD
3 Nov	Dependent types	TBD	TBD	TBD
10 Nov	Module systems	TBD	TBD	TBD
17 Nov	Verified compilation	TBD	TBD	TBD
24 Nov	Partial evaluation	TBD	TBD	TBD
1 Dec	Program synthesis	TBD	TBD	TBD

How to get high marks in this class

How to get a high mark in an essay

Structure

High marks



Low marks

PL

Languages

Processing

Essay marks are awarded for *understanding*,
for *insight and analysis*,
and for *writing quality*.

Essays should be around 1500 words.

1. Contextualise *widely*
2. Analyse *deeply*
3. Appraise *thoughtfully*
4. Elucidate *carefully*
5. Describe *originally*
6. Synthesise *insightfully*
7. Expound *illustratively*
8. Write *stylishly*

Structure

High marks



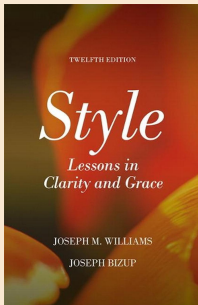
Low marks

PL

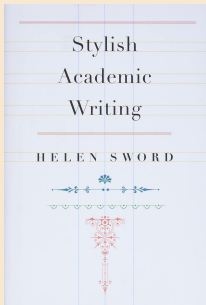
Languages

Processing

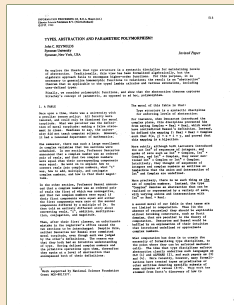
Read a book



Read another book



Read some papers



How to get a high mark in a presentation

Structure

High marks



Low marks

PL

Languages

Processing

Presentation marks are awarded for *clarity*,
for *effective communication*,
and for *selection and organisation of topics*

- | | |
|---------------------------------------|----------------------------------|
| 1. <i>engage</i> with the audience | 4. explain the <i>problem</i> |
| 2. <i>empathize</i> with the audience | 5. bring out the <i>key idea</i> |
| 3. <i>bring people along</i> | 6. have one <i>key example</i> |

Structure

High marks



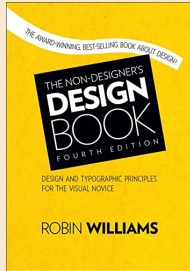
Low marks

PL

Languages

Processing

Read a book



Look at some slides



Watch a presentation



How to get low marks in this class

How to get a low mark in an essay

Structure

High marks

Low marks



PL

Languages

Processing

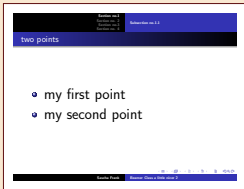


1. be exclusively *critical*
2. *quote* extensively
3. assert *without evidence*
4. stay *vague* and *noncommittal*

How to get a low mark in a presentation

Structure

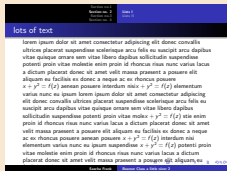
1. read your slides



2. overrun



3. stuff your slides



4. disregard structure

lorem ipsum dolor sit amet consectetur adipiscing elit donec convallis ultrices placerat suspendisse scelerisque arcu felis eu suscipit arcu dapibus vitae quisque ornare sem vitae libero dapibus sollicitudin suspendisse potenti proin vitae molestie enim proin id rhoncus risus nunc varius lacus a dictum placerat donec sit amet velit massa praesent a posuere elit aliquam eu facilisis ex donec a neque ac ex rhoncus posuere aenean posuere interdum nisi elementum varius nunc eu ipsum suspendisse x + y^2 = f(x) potenti proin vitae molestie enim proin id rhoncus risus nunc varius lacus a dictum placerat donec sit amet velit massa praesent a posuere elit aliquam, eu

High marks

Low marks

PL

Languages

Processing

Programming languages: themes

Structure

High marks

Low marks

PL



Languages

Processing

Q: what *is* a program?

Structure

High marks

Low marks

PL



Languages

Processing

Q: what *undecidable question*
are we approximating?

Structure

High marks

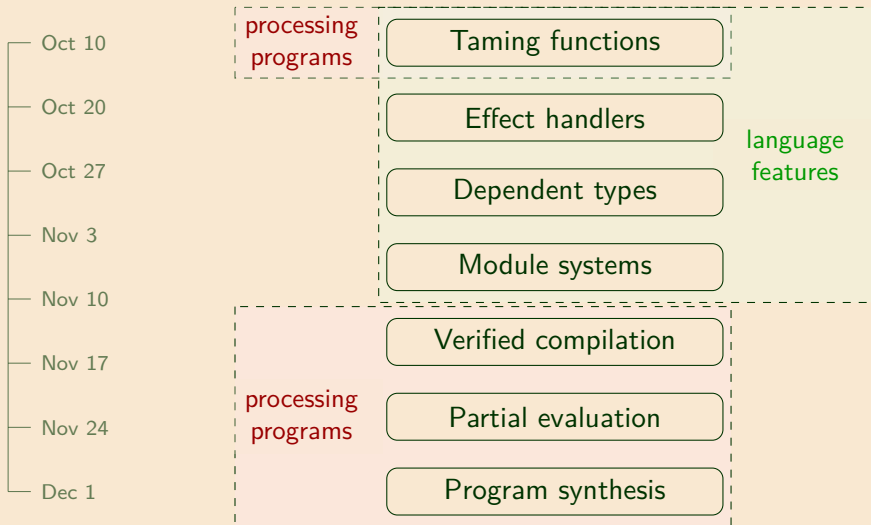
Low marks

PL



Languages

Processing



Programming language features

Structure

High marks

Low marks

PL

Languages



Processing

Question: How can we effectively deal with higher-order functions?

A program is a computation that may treat functions as data

$\lambda x.e$

— ANF —→	all computation unnested
— CPS conversion —→	all calls tail calls
— closure conversion —→	all functions closed
— defunctionalization —→	all functions enumerated
— lambda lifting —→	all functions top-level

Structure

High marks

Low marks

PL

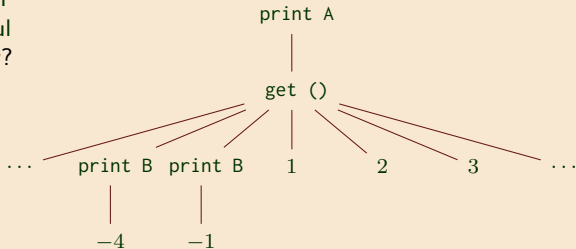
Languages



Processing

Question: How can we extend programming languages with operators that allow powerful manipulation of control flow?

A program is a computation tree with uninterpreted labels



More questions:

Should the effects a program performs be reflected in its **types**?

What new types of programs can be expressed using effect handlers?

What is the connection with **delimited continuations**?

Structure

High marks

Low marks

PL

Languages



Processing

Question: How can we build a powerful, usable, and efficient programming language out of type theory?


$$m < n \Rightarrow n \neq 0 : m < n \rightarrow n \neq 0$$
$$m < n \Rightarrow n \neq 0 \ (s \leq s \ m \leq n) \ ()$$

A program is a blend of logic and computation.

What's undecidable? Type equivalence is undecidable in general



More questions:

How should we handle equality?

How might we write programs in a dependently-typed language?

How might we compile programs effectively?

Structure

High marks

Low marks

PL

Languages



Processing

Question: How might language design support assembling systems from well-specified components?

A program is a large modular system assembled from separately-defined components.

```
module type SET =  
sig  
  type t  
  type elem  
  val empty : t  
  val add : elem → t → t  
  val mem : elem → t → bool  
end
```

```
module MakeSet (Elem: ORDERED) :  
  SET with type elem = Elem.t
```

More questions:

How can we support **abstraction** and **flexible composition**?

What might a **core language** of modules look like?

How might we add support for **recursion**, **higher-order modules**, and **first-class modules**?

What **problems** might arise in sophisticated module systems?

Processing programs

Verified compilation

Structure

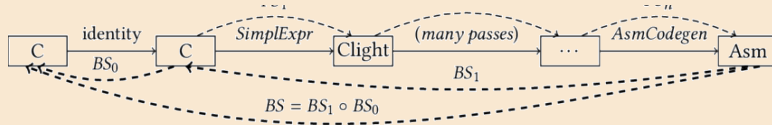
High marks

Low marks

PL

Languages

Processing



Question: How can we preserve semantics when translating a program?

A program is an object with a rich meaning that compilation preserves

What's undecidable? Whether two programs have the same meaning

Partial evaluation

Structure

High marks

Low marks

PL

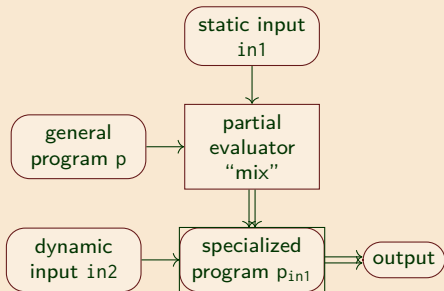
Languages

Processing

Question: How might we perform as much computation as possible in advance?

A program is an *open term* that can be simplified using reductions.

What's undecidable? Whether a program is optimally partially evaluated is undecidable.



More questions:

How can we transform a program to improve its partial evaluation?

Is partial evaluation *useful* in practice?

How can we incorporate equations other than β ?

Structure

High marks

Low marks

PL

Languages

Processing

Question: How can we generate programs from specifications?

A program is an object in a very large search space.

What's undecidable? Whether a program meets a specification is undecidable in general.

