#### Undergraduate Teaching Associates (UTAs)

 Students (and some PhD students) who have agreed to help run small group tutorials with the Teaching Program

• PPTs

- You have an 'A' on the degree so far
- You have an 'A' in first year programming
- For PMTs
  - You have an 'A' on the degree so far
  - You have an 'A' in PMT courses

## **Structure of Introduction**

- General Role of UTAs
- How PPT/PMT fits into teaching/learning
- Use of CATE to enter marks/attendance
- Use of CATE to copy exercises/solutions (PMT)
- Using tutorial pages to make payment claims (sza)
- Introduction to PMT (imh)

#### Undergraduate Teaching Associates (UTAs)

- You were recently first year students
- You have acquired thorough background knowledge of Computer Science
- You have a good understanding of student learning
- You can help first year students develop skills
- As a UTA you should develop
  - Communication skills
  - Debugging skills
- You can participate in the life of the department

### General Role of UTAs

- Two ideas
  - Using students to help with teaching
  - Moving away from a marks based culture
  - Towards active learning (based on student perspective)
- UTAs are closer to learning styles of first year students
- UTAs can gain experience in teaching/communic
- UTAs can gain experience of department

## Helping with teaching

- As first year students
- What did you think was
  - Good?
  - Bad?
  - Could be improved?
- Now you can take an active role
- You can
  - Help in PPT sessions
  - Help in PMT sessions

#### Undergraduate Teaching Associates (UTAs)

- Some will help with PMT meetings
- Some will help with PPT meetings
- Arrangements a bit different for PPT/PMT
  - PPT material distributed as hard copy
  - PMT material available via CATE
  - PMT submissions as hard copy via SAO
  - PPT submissions in electronic form
  - PPT autotested
  - PMT not autotested
  - Both submit marks attendance using CATE

- Both submit claims using TSC

# Main role of (PPT) UTAs

- Helping students to learn how to program
  - Supports programming lecture courses
  - Learning of main teaching languages
  - Learn to to think like programmers
- You will help students to learn
  - How to problem solve?
  - How design/structure a solution?
  - What data structure to use?
  - What algorithm to use?
  - How to test and debug?
- How to think like programmers

# Main role of (PMT) UTAs

- Helping students to reason about programs
  - Supports discrete maths lecture courses
  - Logic
  - Discrete Maths
  - Reasoning about Programs
  - Learn to reason about programs
- You will help students to learn how to understand material taught in discrete maths courses
- How to reason about programs using mathematical concepts

## What is involved

- PPT/PMT helpers
  - Marking student submissions
  - Attending weekly PPT/PMT meeting
  - Keeping an attendance record
  - Entering data into CATE each week
- Time committment
  - PPT/PMT helpers 3 hours per week

# Working as a PPT/PMT helper

- Fixed group students
- See them every week
- Work closely with academic (PPT/PMT)
- Plan PPT/PMT sessions
- Key feedback on assignments
- Influence on students to improve
- PPT: students have to pass online tests to continue to second year
- PMT: students have to pass course work/exams to continue to second year

#### PMT: Marking student submissions

- PMTs submitted on Tuesday evening
- 5/7 assessed PMTs each term
- You should collect scripts from PMT tray in the SAO
- Wednesday
- Mark and return by next PMT meeting
- Marked out of 5.
- (usually on Friday at 12.00/14.00)
- Comments on script vital

- Need to encourage students
- Students need to know how they will do in courseworks/exams
- Help students improve

#### **PPT:** Marking student submissions

- Labs submitted on Monday evening
- You will receive autotest results by email
- Tuesday morning
- You should print these out
- Mark and return by next PPT meeting
- At an agreed time (during lab good)
- Comments on script vital
- Need to encourage students

- Students need to know how they will do in online tests.
- Help students improve

#### **PPT:** Marking student submissions

- Ten marks
  - Correctness and Testing
  - Design, Style and Readability
- (see FAQ + notes)
- autotest NOT automark
- Which tests failed look at code
- Use judgement about seriousness of error
- Try to reverse engineer programming process
- Was program properly tested

#### **PPT:** Marking student submissions

- Design, Style and Readability
- Copious and thorough (if possible)
- Praise good points
- Criticize weak points
- Deduct marks if student doesn't take on previous corrections
- Key idea: to get the student to reflect on programming process and improve
- Note: importance of online tests.

# Helping with PMT meetings

- Meeting timetabled Friday at 12.00/14.00
- If attending optional course agree another time with students/PMT
- Discuss organisation of meeting with PMT
- At meeting:
  - Return marked work
  - Discuss any problems with previous lab
  - Discuss any problems with current lab
- Important that students understand feedback

- Make sure students understand current exercise
- Input attendance/marks into CATE each week

# Helping with PPT meetings

- Arrange time with PPT and students
- Discuss organisation of meeting with PPT
- At meeting:
  - Return marked work
  - Discuss any problems with previous lab
  - Discuss any problems with current lab
- Important that students understand feedback
- Make sure students understand current lab
- Input attendance/marks into CATE

## Helping with PPT meetings

- If possible get students to participate
- If possible ask the students questions
- Can be difficult
- Suggest surgery session for students with difficulty with concepts
- Missing/late work discuss with PPT
- PPT should make decisions about missing/late work
- Inform Peter Cutler (psc) about problems

## **Online Tests**

- Students have to:
  - attend every PPT meeting
  - submit every lab exercise
- The lab marks are zero weighted
- Vital that students complete all labs
- They give crucial feedback on progress
- Students have to pass online tests to pass year
- If students have problems suggest they get help
  - Labs, surgeries

#### **Online Tests**

- Driving Tests
  - Haskell weeks 5 and 6 (term 1)
  - Java/Kenya week 11 (term 1) + resit
  - O-O Java week 10 (term 2)
- Final Lab Exams
  - Haskell week 1 (term 2)
  - Java/Kenya week 11 (term 2)
  - O-O Java week 1 (term 3)
  - Prolog week 7 (term 3)

# **Surgeries and Demonstration**

- Surgeries
  - Twice a week
  - For students with problems understanding concepts taught in lectures
  - Take as long as necessary in session
  - Work with student until they are satisfied that they understand
- Lab Demonstration
  - For students designing and writing programs
  - Put up hand in lab
  - What is problem?
  - See FAQ page

# **Entering Marks/Attendance**

- Start up CATE
- Go to appropriate first year page
- Choose **H** for exercise
- Enter marks
- Choose A
- Enter attendance
- Do this each week
- Requirement for payment

# PMTs Fetching Exercises/Sampl Solution

- CATE provides following
- $\bullet$  Notes click on  ${\bf N}$
- $\bullet$  Exercise click on  ${\boldsymbol{G}}$
- $\bullet$  Sample answer click on  ${\boldsymbol{G}}$
- CATE will be modified to allow this
- Timetable go to PMT homepage

# **Claiming For Payment**

- Use the tutorial system
- Silvana Zappacosta will explain
- Make payment claims weekly
- Enter attendance/marks in CATE

## **Possible Problems**

- Manage time allow an average of 1 hour for marking
- Don't spend too much time on marking there is no need to help outside scheduled times
- Discuss PMT/PPT meeting/marking with PMT/PPT beforehand
- Make sure submissions are marked by PMT/PPT meeting
- Input attendance/marks as soon as possible
- Attend UTA meetings with Peter Cutler (psc) and contact **immediately** him if any questions/problems.