

技能重視の演習及び試験 のオンラインサポート

高塚正浩

School of Computer Science,
Faculty of Engineering



What happened at The University of Sydney

- 2月

- 2月24日 1学期開始。
- 全科目において、オンライン及びF2F (Face to Face)

- 3月

- 国境封鎖
- オンラインでの学期末試験実行環境の構築開始
- 3月23日 (第5週) 全科目オンラインに移行

- 4月

- COVID-19 特別履修取下げ (何千人の単位で海外留学生の数が減少)

What happened at The University of Sydney

- 6月

- 第2-3週 学期末試験。
- 試験監督 (ProctorU Live+/Review+)
- OpenBook ・ 口頭試験

- 7・8月

- 全科目でオンラインを前提とした、履修査定・評価の見直し
- 後期の開始を3週遅らせる。
- 去年の履修者数を上回る。前期に履修取下をした学生の再履修。
- 一部の科目でF2Fの演習授業を許可。(政府より留学生受け入れの特別許可を得るため)

主な問題点

- 新入生はオンライン履修にかなりの戸惑い。
 - 既存の学生はオンライン履修に対してかなりポジティブ
- かなりの学生が孤独を感じる。
 - ❖結果。オンラインTutorなどのサービスの使用頻度が上がる
- ❖教員、指導員からの直接の(F F)指導をありがたがる。
- ❖監督下での試験・技術評価が困難。

オンラインTutorなどのサービスの使用頻度上昇

- 通常ならオンラインTutorサービスは不必要
- 学友とのキャンパス交流が減少 → オンラインサービスの使用。
- 演習レポート、宿題等までもオンラインサービスに頼る。
 - Contract Cheating数の上昇。1年生の科目に500件以上の場合も。
 - オンラインの演習・技術科目において学生たちの学習過程の可視化が必要。
 - Cheating/カンニングの摘発

教員、指導員からの直接の(F F)指導の重要性

- 多くの学生はオンラインツールの使用を受け入れやすい。
- キャンパスでの教員・指導員からのインタラクティブな指導の方が効率的。
- オンラインであっても、実際に同室内にてのインタラクティブな指導を模する方法が必要。

当学科でのTAの経験を基にしたツール

- Edstem.org

- 当学科でTAを務めた卒業生が開発。
- 同期・非同期の協調学習を支援。
- Harvard, Stanford, MITでも今年より採用。

- 以下の問題の解消への手助け
 - ❖ オンラインTutorなどのサービスの使用頻度が上がる
 - ❖ 教員、指導員からの直接の(F F)指導。
 - ❖ 監督下での試験・技術評価が困難。

ed THE UNIVERSITY OF SYDNEY INFO1110 1910 9001 – Discussion

New Thread Search

COURSES

- COMP2017
- CS.next Sandbox
- INFO1110 1910 9001 3

CATEGORIES

- General
- Lectures
- Tutorials
- Challenges
- Problem Sets
- Project
- Software
- Final Exam
- 1910
- Timetable

16 others online

Filter

Pinned

- Welcome to INFO1110!
 Software Cindy Leong STAFF 15h
- Timetable FAQ
 General Victor Kuo STAFF 7d 13 (2 new) 7
- Welcome
 General John Stavrakakis STAFF 14d 3 12

This Week

- Question about Lab
 General Men Men 14h 1
- WSL - Installation Guide and FAQs
 Software Cindy Leong STAFF 15h
- WSL - Where are my files? FAQ
 Software Cindy Leong STAFF 15h
- Question on intensive january for INFO1113
 General Emily Huang 2d 2
- Lab Swap: T16A to T10A, T12A OR T14A
 General Tim Schmid 2d 1

Welcome to INFO1110!

Cindy Leong STAFF 15 hours ago in Software PRIVATE UNPIN STAR WATCH 26 VIEWS

There will be a couple sections to this post. Please read it thoroughly to ensure you are well prepared for this course!

USEFUL LINKS:

Pass requirements: <https://edstem.org/courses/4692/discussion/271741>

Weekly Lessons: <https://edstem.org/courses/4692/discussion/271722>

Problem Sets: <https://edstem.org/courses/4692/discussion/271720>

Final Exam:

- computer exam: <https://edstem.org/courses/4692/discussion/271726>
- oral exam: <https://edstem.org/courses/4692/discussion/271737>

1) Unix system - COMPLETE THIS BY END OF WEEK 1!

In this course, we highly recommend the use of a **Unix** based system.

If you're using macOS or Linux, you're all covered and running Unix already!

Otherwise if you're on Windows, we recommend ONE of the following three options:

- Install the [Windows Subsystems for Linux](#) (only available on Windows 10). Guide here: <https://edstem.org/courses/4692/discussion/273058>
- Install [Git Bash](#).



COURSES

COMP2017

CS.next Sandbox

INFO1110 1910 9001 3

INFO1110 1910 90... 4134

CATEGORIES

 General

 Lectures

 Tutorials

W1

W2

W3

W4

W5

W6

W7

W8

W9

W10

W11

W12

W13

 Evaluation

Filter

1 Jun 2020

 week 6 cone ✓

 Tutorials - W5 Joan Liao 3mth 2
 week 5 cats revision

 Tutorials - W5 Joan Liao 3mth 6

20 Apr 2020

 Weekly helpful recap — notes and ex... ✓

 Tutorials - W5 Samuel Catts 4mth 2
 Memory Allocation (list copy function)

 Tutorials - W5 Philip Law 4mth 1

6 Apr 2020

 Pass by Reference and Value ✓

 Tutorials - W5 Kim Zheng 4mth 11 1

30 Mar 2020

 lab5 q5 part3 ✓

 Tutorials - W5 Yizhen Song 5mth 1
 t5 q1 p4 ✓

 Tutorials - W5 Yibo Zhao 5mth 1
 W5 Q6 ✓

week 5 cats revision

You are viewing this thread in readonly mode because this course has been archived.

Joan Liao CE333 OE315-1 W10E

3 months ago in Tutorials - W5

164

VIEWS

Run

PYTHON



```

1 cats = ['Tom', 'Harvey', 'Spots']
2 i = 0
3 while i < len(cats):
4     currentcat = cats[i]
5     print(currentcat)
6     print(len(currentcat))
7     longestcat = cats[0]
8     print(longestcat)
9     print(len(longestcat))
10    if len(currentcat) > len(longestcat):
11        longestcat = currentcat
12    i += 1
13 print(longestcat)

```

Spots

5

Tom

3

Spots

 Program exited with code 0

Edstem.org : 同期共同作業。

The screenshot displays the Edstem.org workspace interface. At the top, the header includes the Edstem logo, The University of Sydney logo, and the course identifier "INFO1110 1910 9001 - Workspaces". The workspace title is "Lecture - Week 1".

The interface is divided into three main sections:

- File Explorer (Left):** Shows a list of files including "README", "hello.py" (selected), and "jj.py".
- Code Editor (Center):** Displays the content of "hello.py":

```
1 print("Hello!")
2 radius = 100
3
4 print(radius)
5
6 print(str(radius + 50)) //help here!!!
7
8 print(radius)
9 radius = (radius + 50)
10
11
12
13 print(radius)
14
```

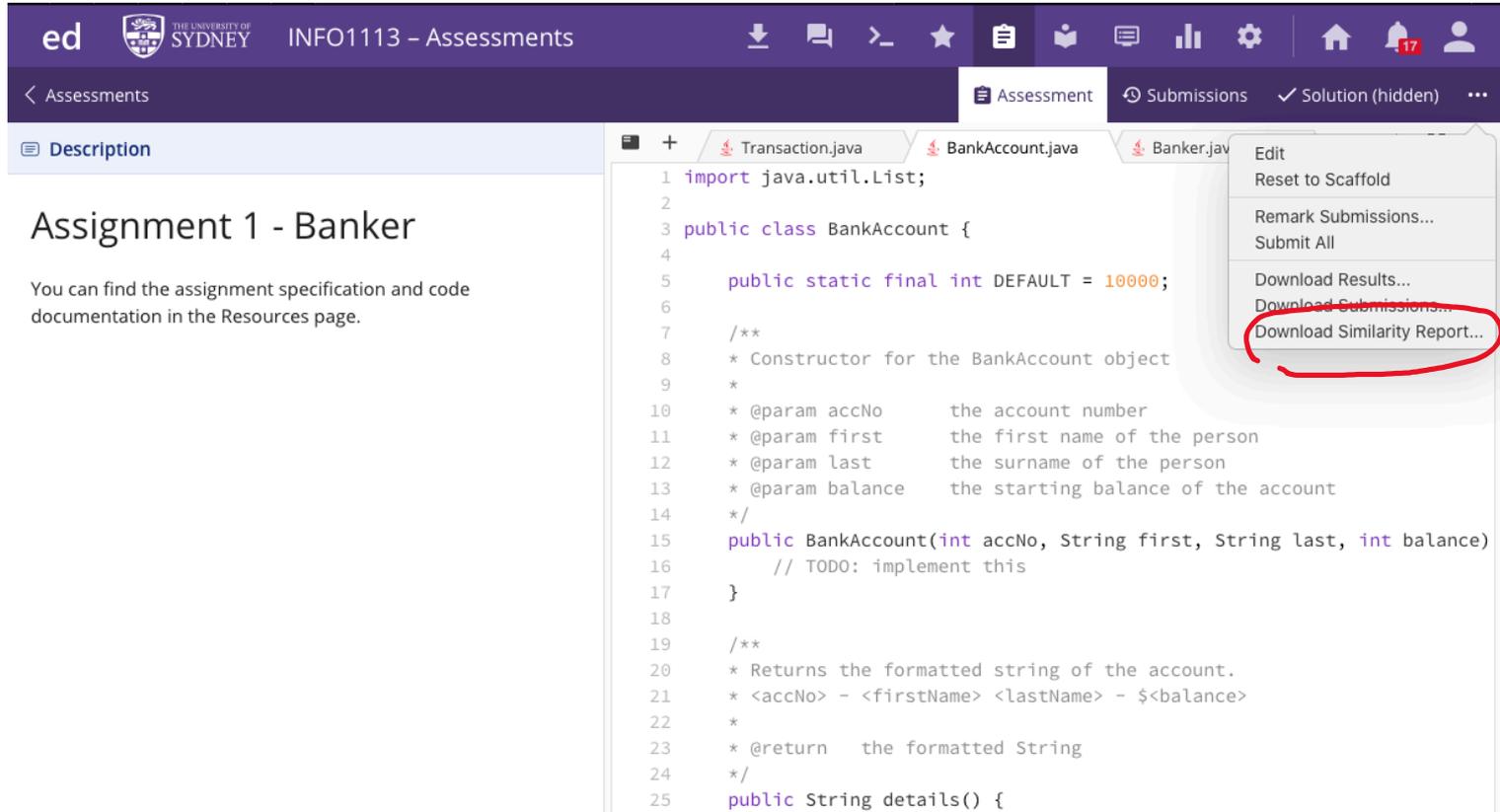
A comment on line 5, "`//help here!!!`", is highlighted with a blue box, and the name "John Stavrakakis" is visible above it, indicating a comment or edit.
- Participants (Right):** Lists two participants: "John Stavrakakis" and "Masahiro Takatsuka".
- Chat (Right):** Shows a chat area with a timestamp "Today at 2:50 PM" and a message "hello" sent by "Masahiro Takatsuka".

Edstem.org : 同期共同作業。

The screenshot displays the Edstem.org workspace interface. At the top, the header includes the Edstem logo, The University of Sydney logo, and the course identifier "INFO1110 1910 9001 - Workspaces". The main workspace area is titled "Lecture - Week 1" and contains a file explorer on the left with files "README", "hello.py", and "jj.py". The "hello.py" file is open in the code editor, showing Python code with a comment and a line of code that has been edited. The chat window on the right shows two participants: John Stavrakakis and Masahiro Takatsuka. A chat message "hello" is visible, sent by Masahiro Takatsuka.

```
1 print("Hello!")
2 radius = 100
3
4 print(radius)
5
6 # comments are done using #, not //
7 John Stavrakakis will add two numbers, convert to string, then p
8
9 print(str(radius + 50)) # help here!!!
10
11 print(radius)
12 radius = (radius + 50)
13
14
15
16 print(radius)
17
```

Edstem.org : 学習過程の可視化 ・ Cheating/カンニングの摘発



The screenshot displays the Edstem.org web application. The top navigation bar includes the 'ed' logo, 'THE UNIVERSITY OF SYDNEY', and 'INFO1113 - Assessments'. Below this, a breadcrumb trail shows 'Assessments' and 'Assessment'. The main content area is divided into two sections: 'Description' and 'Code Editor'.

Description: The title is 'Assignment 1 - Banker'. The text below reads: 'You can find the assignment specification and code documentation in the Resources page.'

Code Editor: The editor shows a Java file named 'BankAccount.java'. The code is as follows:

```
1 import java.util.List;
2
3 public class BankAccount {
4
5     public static final int DEFAULT = 10000;
6
7     /**
8      * Constructor for the BankAccount object
9      *
10     * @param accNo    the account number
11     * @param first    the first name of the person
12     * @param last     the surname of the person
13     * @param balance  the starting balance of the account
14     */
15     public BankAccount(int accNo, String first, String last, int balance)
16         // TODO: implement this
17     }
18
19     /**
20     * Returns the formatted string of the account.
21     * <accNo> - <firstName> <lastName> - ${balance}
22     *
23     * @return    the formatted String
24     */
25     public String details() {
```

A context menu is open over the code editor, listing several actions: 'Edit', 'Reset to Scaffold', 'Remark Submissions...', 'Submit All', 'Download Results...', 'Download Submissions...', and 'Download Similarity Report...'. The 'Download Similarity Report...' option is circled in red.

Edstem.org : 学習過程の可視化 ・ Cheating/カンニングの摘発

Course similarity report

[View graph](#)

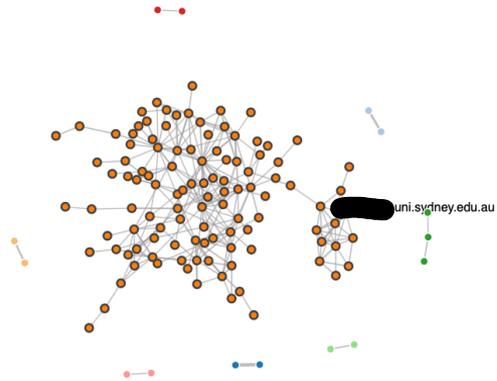
	User 1	User 2	Score	Matched lines	Link
1	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	43.08	545	Link
2	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	29.37	495	Link
3	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	26.13	223	Link
4	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	24.85	344	Link
5	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	24.55	423	Link
6	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	23.31	256	Link
7	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	22.84	65	Link
8	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	22.25	334	Link
9	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	22.09	252	Link
10	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	21.91	508	Link
11	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	21.72	362	Link
12	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	20.30	342	Link
13	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	20.24	50	Link
14	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	20.20	288	Link
15	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	19.96	47	Link
16	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	19.62	346	Link
17	[REDACTED]@uni.sydney.edu.au	[REDACTED]@uni.sydney.edu.au	18.54	277	Link

- 全ての Submission に対して検査可能。
- 演習作業の過程においてもチェック可能。
 - 学習過程のモニタリング。

Edstem.org : 学習過程の可視化 ・ Cheating/カンニングの摘発

```
@uni.sydney.edu.au
715     }
716
717     for (int i : others) {
718         BankAccount mergedAcc = accounts.get(i);
719         if (mergedAcc == null) {
720             throw new Exception("no such account");
721         }
722         if (destination == mergedAcc) {
723             throw new Exception("sender cannot be receiver");
724         }
725         Transaction newTransaction = new Transaction(transacti
726             mergedAcc, mergedAcc.getBalance(), prevHash);
727         newList.add(newTransaction);
728         prevHash = newTransaction.getCurrentHash();
729     }
730     for (Transaction i : newList) {
731         transactions.add(i);
732         i.getSender().processTransaction(i);
733         i.getReceiver().processTransaction(i);
734     }
735     System.out.println("success");
736 } catch (Exception e) {
737     System.out.println(e.getMessage());
738 }
739 }
740
741 /**
742  * Displays the lowest balance in the system.
743  */
744 public void min() {
745     // TODO: implement this
746     List<BankAccount> temp = new ArrayList<>();
747     for (BankAccount i : accounts.values()) {
748         temp.add(i);
749     }
750     if (temp.size() > 0) {
751         int min = BankAccount.findMin(temp);
752         System.out.println("$" + min);
753     } else {
754         System.out.println("no accounts");
755     }
756 }
757
758 /**
759  * Displays the highest balance in the system.
760  */
761 public void max() {
762     // TODO: implement this
763     List<BankAccount> temp = new ArrayList<>();
764     for (BankAccount i : accounts.values()) {
765         temp.add(i);
766     }
767     if (temp.size() > 0) {
768         int max = BankAccount.findMax(temp);
769         System.out.println("$" + max);
770     } else {
771         System.out.println("no accounts");
772     }
773 }
774
775 /**
776  * Displays the lowest balance in the system.
777  */
778 public void min() {
779     // TODO: implement this
780     List<BankAccount> newAccount = new ArrayList<>();
781     for (BankAccount i : accounts.values()){
782         newAccount.add(i);
783     }
784     if(newAccount.size()>0){
785         int min = BankAccount.findMin(newAccount);
786         System.out.println("$" + min);
787     }else{
788         System.out.println("no accounts");
789     }
790 }
791
792 /**
793  * Displays the highest balance in the system.
794  */
795 public void max() {
796     // TODO: implement this
797     List<BankAccount> newAccount = new ArrayList<>();
798     for (BankAccount i : accounts.values()){
799         newAccount.add(i);
800     }
801     if(newAccount.size()>0){
802         int max = BankAccount.findMax(newAccount);
803         System.out.println("$" + max);
804     } else {
805         System.out.println("no accounts");
806     }
807 }
808
809 for(int i : others){
810     BankAccount mergeAcc = accounts.get(i);
811     if(mergeAcc == null){
812         throw new Exception("no such account");
813     }
814     if(destAcc == mergeAcc){
815         throw new Exception("sender cannot be receiver");
816     }
817     Transaction newTrans = new Transaction(transactions.si
818         newList.add(newTrans);
819         prevHash = newTrans.getHash();
820     }
821     for(Transaction i : newList){
822         transactions.add(i);
823         i.getSenderAcc().processTransaction(i);
824         i.getReceiverAcc().processTransaction(i);
825     }
826     System.out.println("success");
827 }catch(Exception e){
828     System.out.println(e.getMessage());
829 }
830 }
```

Edstem.org : 学習過程の可視化



Cheating/カンニングの摘発

- 演習作業の過程においてソーシャルネットワークの形成をモニタリング。
- 早期発見、対応が可能。

User 1	User 2	Score	Matched lines	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	26.13	223	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	24.55	423	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	23.31	256	Link
[Redacted]	[Redacted]@sydney.edu.au	22.84	65	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	22.25	334	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	22.09	252	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	21.91	508	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	21.72	362	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	20.3	342	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	20.24	50	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	20.2	288	Link
[Redacted]	[Redacted]@uni.sydney.edu.au	19.96	47	Link

監督下での試験・技術評価が困難

- 前期の学期末試験には、オンラインの試験監督サービスを使用。
 - 色々なカンニングが可能。
- OpenBook筆記試験と口頭試験の併用。

OpenBook筆記試験と口頭試験の併用

- Edstem.orgを用いた、OpenBook筆記試験・提出。
- 自動採点（Edstem.orgを使用）
- 各自20分程度の口頭試験（主にZoomを使用）
 - 筆記試験の回答を基に質問
 - その場で追加質問し回答を変更させる。

OpenBook筆記試験と口頭試験の併用

- 1年のプログラミング演習
- 学生1000人、指導員53人。
- 指導員の、口頭試験の品質管理
 - 事前に典型的な回答例を基にどのような様な口頭質問するかを指導員に教授

OpenBook筆記試験と口頭試験の併用

```
e = OSError('File does not exist :(')
print(isinstance(e, FileNotFoundError))
```

```
e = ZeroDivisionError()
print(isinstance(e, ArithmeticError))
```

```
e = OSError()
raise e
print(isinstance(e, OSError))
```

What is the output of the code above when executed in Python3?

- If no response or the student seems completely lost: Ask "Does this compile?"

- If the student says "I don't know what isinstance is", tutor can brief them on what it's supposed to do.

- Tutor can clarify small details, such as what is `raise`?

1 mark: The first print statement prints `False`

1 mark: The second print statement prints `True`

1 mark: For `raise` will stop the program and last print statement will not run.

Follow up question: if `raise e` was not there, what would be the output?

1 mark: The third print statement prints `True`

後期・F2Fへの対応

- 授業はオンラインだが、限られた演習においてF2F。
- F2Fを履修する学生は、学生証をスキャンしてビルにアクセス
- コンピュータ演習室入室をアプリで記録。

- 大学の教室内では、ソーシャルディスタンスのルール適用は免除。
- 一旦部屋を退出するとルール適用。(矛盾?)

- 手洗い、消毒液、アルコールワイプなどビル内の各所に設置。
- 教員・指導員へのマスクの提供。

Q&A