COMP1400
Programming for Designers
Hailun Tan
Contacts

Lecturer in Charge: Hailun Tan

Web pages:
www.cse.unsw.edu.au/~cs1400

Email: cs1400@cse.unsw.edu.au

Tutor(s): TBA
Class announcements

Announcements will be made in lectures and posted on the class web pages.

It is your responsibility to keep up to date with announcements.

Web pages:

www.cse.unsw.edu.au/~cs1400
Labs

Programming can only be learned by doing.
Lab exercises will be posted on the web pages for each subject.

Labs start in Week 2.

Labs:

Assessment

• Lab mark (10%)
• Three assignment tasks (10% each)
• Mid-semester practical exam (10%)
• A practical exam in Week 13 (20%)
• A written exam (30%)
• 40% in the final exams (practical and written) required to pass
Java and BlueJ

This course will be taught using the Java programming language and the BlueJ interactive development environment.

Java and BlueJ can be downloaded for free at: http://www.bluej.org/
Textbook

Objects First with Java: A Practical Introduction using BlueJ
David J. Barnes & Michael Kölling
Pearson, 6th edition 2016
http://www.bluej.org/objects-first
What is a computer?

• It is an incomplete machine
• computer hardware only provides the building blocks for a working machine
• A program assembles the blocks and says how they will be used and in what order
What’s the difference?
A program is a sequence of instructions that make a machine.
What does this do?

private static int min(int[] nums)
{
    int m = nums[0];
    for (int i = 1; i < nums.length; i++)
    {
        int n = nums[i];
        if (n < m)
            m = n;
    }
    return m;
}
ΣΩΚΡΑΤΗΣ: κατέβηκαν χθες εἰς Πειραιᾶ μετὰ Γλαύκωνος τοῦ Ἀρίστωνος προσευξόμενος τε τῇ θεῷ καὶ ἄμα τῇ ἔορτῇν βουλόμενος θεάσασθαι τίνα τρόπον ποιήσουσιν ἄτε γὰν πρῶτον ἄγοντες. καλὴ ἡ μὲν μέντοι θεὰν ἐφαίνετο πρέπειν ἢν οἱ Θρᾶκες ἔπεμπον. προσευξόμενοι δὲ καὶ Θεωρήσαντες ἀπῆμεν πρὸς τὸ ἄστυ. κατιδὼν οὖν πόρρῳδεν ἡμᾶς οἶκας ὡρμημένοις Πολέμαρχος ὁ Κεφαλὸς ἐκέλευσε δραμόντα τὸν παῖδα περιμεῖναι ἐν κελεύσαι. καὶ μοῦ ὁπισθὲν ὁ παῖς λαβόμενος τοῦ ἰματίου, κελεύει ύμᾶς, ἐφη, Πολέμαρχος περιμεῖναι.
private static int min(int[] nums) {
    int m = nums[0];
    for (int i = 1; i < nums.length; i++) {
        int n = nums[i];
        if (n < m) {
            m = n;
        }
    }
    return m;
}
...is the same as this...

def min(nums):
    m = nums[0]
    for i in range(0,length(nums)):
        if (nums[i] < m):
            m = nums[i]
    return m
...is the same as this...

def min(nums):
    m = nums[0]
    for i in range(0,length(nums)):
        if (nums[i] < m):
            m = nums[i]
    return m

...and this...

min([X], X).
min([X | Xs], X) :-
    min(Xs, Y),
    X < Y, !.
min([_ | Xs], Y) :-
    min(Xs, Y).
...is the same as this...

def min(nums):
    m = nums[0]
    for i in range(0,length(nums)):
        if (nums[i] < m):
            m = nums[i]
    return m

...and this...  ...and this.

min([X], X).
min([X | Xs], X) :-
    min(Xs, Y),
    X < Y, !.
min([_ | Xs], Y) :-
    min(Xs, Y).

LDA $0,X
LOOP: CMP $0,X
BCC SKIP
LDA $0,X
SKIP: DEX
BNE LOOP
RTS
Procedural thinking

Programming is about thinking in terms of step-by-step procedures to achieve an outcome.

Mastering procedural thinking is more important than any particular programming language.
Programming is like Pancakes

1. Put ingredients in bowl.
2. Beat until smooth.
3. Heat frying pan.
4. Pour 1/4 cup of mixture into pan.
5. Wait for bubbles.
6. Flip.
7. Wait 30s or until brown underneath.
8. Transfer to plate.
9. Repeat steps 4-8 until all mixture is used.
Computers are stupid

To put ingredients in bowl:

1. Take bag of flour from cupboard
2. Take bottle of milk from fridge
3. Take carton of eggs from fridge
4. For each person served:
   1. Transfer 1 cup of flour from bag to bowl
   2. Transfer 1 cup of milk from bottle to bowl
   3. Remove 1 egg from carton
   4. Crack egg and pour contents into bowl
   5. Put eggshell in compost bin
void addIngredients(Bowl bowl, int numberOfPeople)
{
    Flour flour = theCupboard.get("Flour");
    Milk milk = theFridge.get("Milk");
    List<Egg> eggs = theFridge.get("Eggs");

    for (int i = 0; i < numberOfPeople; i++)
    {
        flour.transfer(1, bowl);
        milk.transfer(1, bowl);
        Egg egg = eggs.remove(0);
        egg.crackInto(bowl);
        egg.dispose();
    }
}
The 6 Stages of Programming

1. Requirements  What do they want?
2. Specification  What should it do?
3. Design        How will it work?
4. Implementation How is it made?
5. Testing       Does it actually work?
6. Debugging     What went wrong? How to fix?
The essential skills of programming

Procedural thinking: knowing how to write a sequence of instructions to achieve an outcome.

Language literacy: knowing how to read and write code.

Testing and debugging: knowing how to test your code and track down bugs.
The 7th Stage of Programming

1. Requirements
2. Specification
3. Design
4. Implementation
5. Testing
6. Debugging

7. Documentation
Good code

A well-written computer program:

1. Tells the computer what to do,

2. Tells another human being what the computer is going to do.
void AddIng(Bowl b, int n) {
    Flour f = c.Get("Flour");
    Milk m = fr.Get("Milk");
    List<Egg> e = fr.Get("Eggs");
    for (int i=0; i<n; i++)
    {
        f.Put(1,b);m.Put(1,b);
        Egg el=e.rem(0);
        el.cr();el.Put(b);
        el.Disp();
    }
}
void AddIngredients(Bowl bowl, int numberOfPeople)
{
    // 1 cup of SR flour per person
    Flour flour = theCupboard.get("Flour");
    flour.transfer(numberOfPeople, bowl);

    // 1 cup of milk per person
    Milk milk = theFridge.get("Milk");
    milk.transfer(numberOfPeople, bowl);

    // 1 egg per person
    List<Egg> eggs = theFridge.Get("Eggs");
    for (int i = 0; i < numberOfPeople; i++)
    {
        Egg egg = eggs.remove(0);
        egg.crackInto(bowl);
        egg.dispose();
    }
}
Style Guide

http://www.bluej.org/objects-first/styleguide.html