CMPT/ISYS 140/141/143/145: Introduction to Programming

8 Sep 2005
CMPT14x
Dr. Sean Ho
Trinity Western University

Please sit up front and pick up a copy of the syllabus (1 page double-sided)

TRINITY WESTERN UNIVERSITY

http://cmpt14x.seanho.com/

Outline for today

- Course information
  - Course website
  - Syllabus
  - Schedule
- Programming as problem-solving
  - Tools, toolsmiths, toolboxes
  - Top-down vs. bottom-up design
  - Example: woodcutting
- Demo of our Modula-2 programming environment

About the 14x series

- Everyone meets MWF 14:35-15:50
  - 140,145: also meet R 08:00-09:15
- 141,143 run the whole semester
  - 140 runs the first six weeks only
  - 145 runs the last six weeks (but see assignments)
- Credit hours: 140=3, 141=4, 143=2, 145=2
- The usual sequence for most students is 140+145 (total of 5 credits), unless you're not planning to go further.

Course website

- http://cmpt14x.seanho.com/
- Also linked from eCourses

- Note exam chs1-7 on W-Th 26-27Oct:
  - All attend (even those who don't regularly attend Th section)
  - This serves as the final for CMPT140

Lab sections

- Three 4hr sections: MTW 6-10pm (sign up)
- At the main computing lab in Neufeld9
- Turn in your labs to your TA during your lab section
- Feel free to work in the lab at any open time
  - You have priority over other students when you're doing CMPT classwork
- Non-14x lab assistants are not prepped to answer your 14x questions
  - But they can handle printing problems, etc.

Outline for today

- Course information
  - Course website
  - Syllabus
  - Schedule
- Programming as problem-solving
  - Tools, toolsmiths, toolboxes
  - Top-down vs. bottom-up design
  - Example: woodcutting
- Demo of our Modula-2 programming environment

The Art of the Toolsmith

- Computers and software are tools; Computing scientists are toolsmiths
- The success of the tool is evaluated by the user, not by the toolmaker!

```
+   threadfn = create->threadfn;
+   data = create->data;
+
+   /* Block and flush all signals (in case we're not from keventd). */
+   sigfillset(&blocked);
+   sigprocmask(SIG_BLOCK, &blocked, NULL);
+   flush_signals(current);
+
+   /* By default we can run anywhere, unlike keventd. */
+   set_cpus_allowed(current, mask);
+
+   /* OK, tell user we're spawned, wait for stop or wakeup */
+   __set_current_state(TASK_INTERRUPTIBLE);
+   complete(&create->started);
+   schedule();
+
+   if (!kthread_should_stop())
+      ret = threadfn(data);
+
+   /* It might have exited on its own, w/o kthread_stop.  Check. */
+   if (kthread_should_stop()) {
+      kthread_stop_info.err = ret;
+      complete(&kthread_stop_info.done);
+   }
+   return 0;
```

$\neq$

"the code is so beautiful!"

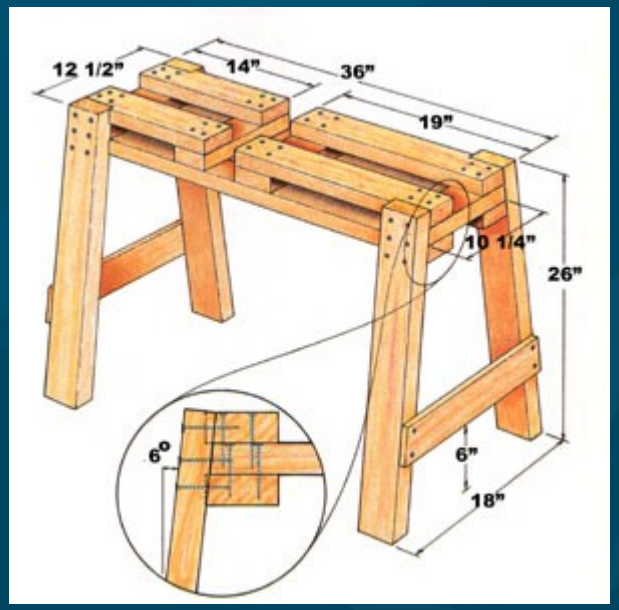
"does it do the job?"

The right tool for the right job

- Beware the technological imperative: when technology drives us instead of us driving technology
- Ask "what tools are appropriate for this problem?"
- Examples of problems suitable for computers? Problems *not* suitable for computers?
- "If all you have is a hammer, everything looks..."

Toolchains

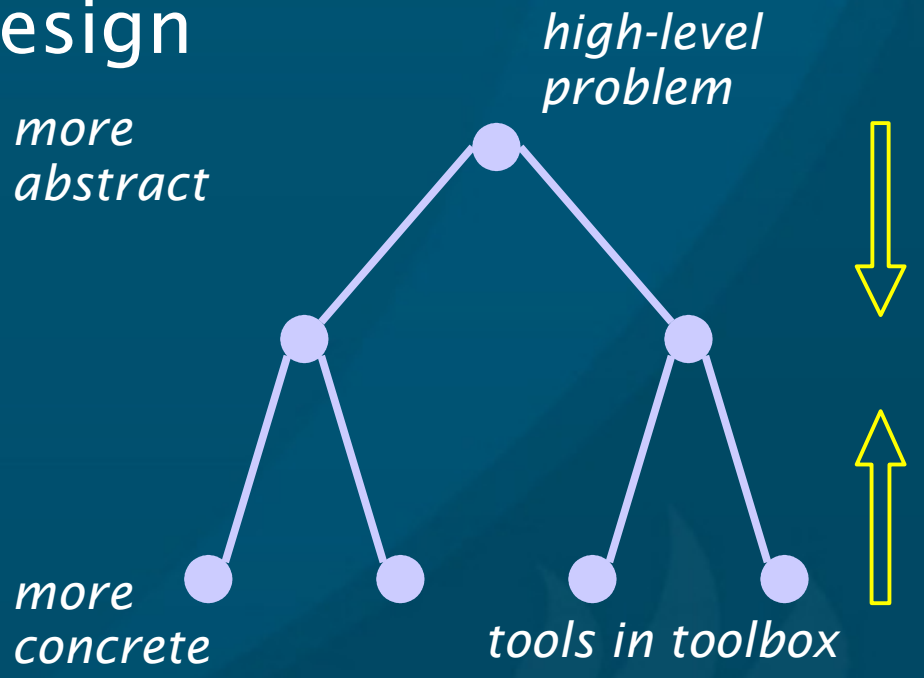
- Complex problems need sophisticated tools
- Complex tools are built up from simpler tools
- Always know what's in your toolbox:
  the tools you have to tackle problems
  - In software: libraries
  - In math: axioms
  - In philosophy: worldview, context



- In 14x: Modula-2 + libraries

Problem solving

- Top-down vs. bottom-up design

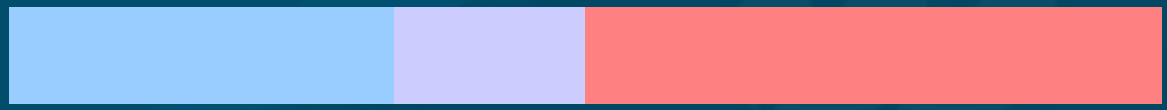


- ***W***rite everything down
- ***A***pprehend the problem
- ***D***esign a solution
- ***E***xecute the plan
- ***S***crutinize the results

Designing software vs. "hacking code"

- Look before you leap; think before you speak; ***design*** before you code!
- Programmer's optimistic schedule:
  - 4/5th coding
  - 1/5th testing/debugging

- Real-life schedule:
  - 1/3rd planning (***W***rite, ***A***pprehend, ***D***esign)
  - 1/6th coding (***E***xecute)
  - 1/2 testing/debugging (***S***crutinize)

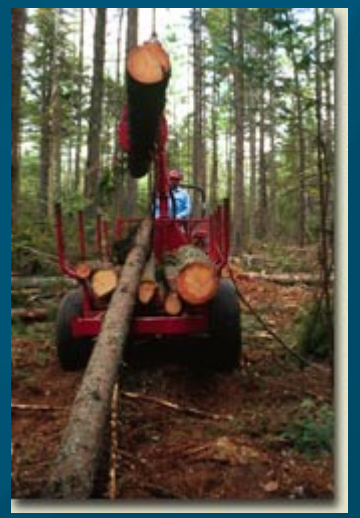


Write
Apprehend
Design
Execute
Scrutinize

Woodcutting example

- (see overheads / text pp.4-5)
- What are the library functions used in each version?

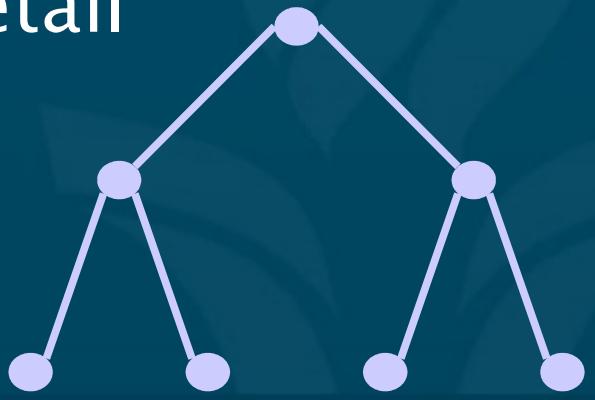
Woodcutting example



- We write out the solution in different levels of detail depending on
  - Who/what is executing the solution
  - What tools are available
- The solution is different for
  - An experienced lumberjack with good tools
  - A rookie who's never used a chainsaw
  - A busy construction foreman
  - A software-controlled robot
- *(which are more abstract / more concrete?)*

Keywords from today (1.1-1.4)

- Toolsmiths must know their toolboxes
  - (what does it mean for a computing scientist to be a toolsmith?)
- Top-down vs. bottom-up
- First step in problem-solving? (don't code yet!)
- ***WADES*** *(Write, Apprehend, Design, Execute, Scrutinize)*
- Levels of abstraction / levels of detail



Outline for today

- Course information
  - Course website
  - Syllabus
  - Schedule
- Programming as problem-solving
  - Tools, toolsmiths, toolboxes
  - Top-down vs. bottom-up design
  - Example: woodcutting
- Demo of our Modula-2 programming environment

Stonybrook demo

- (demo of the Modula-2 programming environment)

Why Modula-2?

- Why not Java, C++, C#, PHP, Ruby, etc.?
- Syntax vs. semantics (more in a later section)
- At the CMPT14x level, the semantics of procedural programming in all these languages are pretty much the same
  - The only difference is syntax:
    VAR c : CHAR; (Modula-2)
    char c; ©
- After this class, you'll be able to pick up any procedural language pretty quickly

Outline for today

- Course information
  - Course website
  - Syllabus
  - Schedule
- Programming as problem-solving
  - Tools, toolsmiths, toolboxes
  - Top-down vs. bottom-up design
  - Example: woodcutting
- Demo of our Modula-2 programming environment

TODO items

- Pick up a copy of the syllabus
- Sign-up for a lab section (MTW 6-10pm)
- Buy 14x coursepack (vols1-2) from bookstore
  - Or borrow from a previous CMPT14x student
  - Read 1.1-1.7 for tomorrow
- Go to Neu9 computer lab:
  - Make sure you can login
  - Stonybrook intro on course www (due 14Sep)
- Ch1 quiz next Monday start of class