What is Computer Science?

What is it that distinguishes it from the separate subjects with which it is related? What is the linking thread which gathers these disparate branches into a single discipline? My answer to these questions is simple — it is the art of programming a computer.

C.A.R. (Tony) Hoare

Programming ≠ Computer Science

What is the nature of intelligence? How can one predict the performance of a complex system? What is the nature of human cognition? Does the natural world 'compute'? It is the interplay between such fundamental challenges and the human condition that makes computer science so interesting. The results from even the most esoteric computer science research programs often have widespread practical impact. Computer security depends upon the innovations in mathematics. Your Google search for a friend depends on state-of-the-art distributed computing systems, algorithms, and software. Your telephone call is transmitted using the mathematics, design patterns, and libraries of computer science.

What is efficient design? What are programs, code, object-oriented design, patterns, idioms, design patterns, libraries? What toolkits do you bring to software development? Mathematics, design patterns, libraries — standard and Duke CPS.

Efficient design, programs, code

And artificial intelligence.

CPS 1001.2

Program Design and Analysis II

Professor: Jeffrey Forbes

M, W 4:25-5:40
B101 LRC
CPS 100

Program Design and Analysis II

Welcome!
Course Overview

Lectures, Recitations, Quizzes, Programs

Recitation based on questions given out in previous week

Discuss answers, answer new questions, small quiz

More opportunities for questions to be answered.

Recitation based on readings, questions, programs

Online quizzes used to motivate/ensure reading

In-class questions used to ensure understanding

Programs

Theory and practice of data structures and OO programming

Fun, practical, tiring...

Weekly programs and longer programs

Exams/Tests

Semester: closed book

Final: open book

Frequently Asked Questions

What is the prerequisite? (choose one)

CPS 6

4 or 5 on AP Computer Science AB exam

What about CompSci 100E?

You should take 100E if you took Engr 53 and do not have Java experience

How does this course fit into the curricula?

Required for majors & minors

Solid grounding in programming, data structures, and algorithms

It's the foundation of your whole computer science education.

What is recitation? Is it required? When does it start?

Recitation is a more hands-on section where you will do problems and discuss solutions. Your work will be graded.

Recitation begins tomorrow

How do I keep up to date?

Read web page regularly

http://www.cs.duke.edu/courses/spring05/cps100

Read discussion forum regularly


Read your email

On the subject of questions...

Did you ask any good questions today?

Ideas and Information

by Nobel prize winning physicist Arno Penzias

Questions which illuminate help nourish ideas

Children are born curious

Fear of public displays of ignorance prevents learning
CPS 1001.9

Practice questions

What year are you?
1. Frosh
2. Soph
3. Junior
4. Senior
5. Grad

What kind of computer do you own/use most often?
1. Windows PC
2. Mac
3. Linux box
4. Other
5. What’s Google?
6. Googleplex
7. > 100 billion
8. > 1 trillion
9. > 10 trillion
10. > 1 googol

How many web pages does Google index?
1. < 10 million
2. < 100 million
3. < 1 billion
4. < 10 billion
5. < 100 billion
6. < 1 trillion
7. > 1 trillion
8. A googolplex
9. What’s Google?
10. How many web pages does Google index?

CPS 1001.10

Tradeoffs
Simple, elegant, quick, efficient: what are our goals in programming?

What does XP say about simplicity? Einstein?

How do we decide what tradeoffs are important?
Tension between generality, simplicity, elegance, CPU time, runtime, space, your alll programs, run anywhere.

Fast programs, small programs, run anywhere. How do we decide what

Tradeoffs...
Java has a huge standard library organized in packages: `java.lang`, `java.util`, `javax.swing`, … API browseable online, but Eclipse IDE helps a lot.

Java methods have different kinds of access: 
- Public methods
- Private methods
- Protected and Package methods

Primitive types (int, char, double, boolean) are not objects but everything else is literally an instance of class `Object`.

foo.callMe();

Arrays are typed and fixed in size when created.

Not like objects from `std::vector`. Can store int, double, string, foo.

Don't have to fill the array, but cannot expand afterwards.

Ensure correct order in C++/Tapestry.

Arrays are typed and fixed in size when created.

Basic data structures and algorithms

Java has a huge standard library.

What does `try` do? Why is it wrong?

Both: Use either than re-implement but know how to do.

Java.net package has lots of data structures and algorithms.

Array lists offer to themselves internally.

Accessing elements can require a downcast.

Stores objects not primitives.

Arrays (and related class Vector and interface List) grows intelligently.

Java.util package has lots of data structures and algorithms.

Search for the word? Avoid counting twice?

Do search engines do this? Do they change the priority of a webpage based on search engines?

Downside of this approach for search engines?

... Constraints on solving this problem

We must read every word in the file (or web page).

Where should we look for data structure and algorithmic improvements?

Where do we look for data structure and algorithmic improvements?

Why is it wrong? 

...
Search: measuring performance

How fast is fast enough?

```java
/**
 * pre: a contains a.size() entries
 * post: return true if and only if key found in a
 */
boolean search(ArrayList a, String key)
{
    for(int k=0; k < a.size(); k++)
        if (a[k].equals(key)) return true;
    return false;
}
```

Java details: parameters? Return values? ArrayLists?

How do we measure performance of code? Of algorithm?

Does processor make a difference? G5? Itanium? 64-bit?

Tradeoffs in processing and counting

Read words, then sort, determine # unique words?

frog, frog, frog, rat, tiger, tiger, tiger, tiger

If we look up words one-at-a-time and bump counter if we haven't seen a word, is this slower than previous idea?

How do we look up word, how do we add word?

Are there kinds of data that make one approach preferable?

Why can we pass different kinds of objects to test?

```java
...; System.out.println(System.currentTimeMillis()-start);
```
Example of inheritance

What is behavior of a shape?
void doShape(Shape s) {
    System.out.println(s.area());
    System.out.println(s.perimeter());
    s.expand(2.0);
    System.out.println(s.area());
    System.out.println(s.perimeter());
}

Shape s1 = new Circle(2);
Shape s2 = new Square(4);
Shape s3 = new Rectangle(2,5);
doShape(s1);
doShape(s2);
doShape(s3);

Inheritance (language independent)

First view: exploit common interfaces in programming
Iterators in Java or C++/Tapestry
Implementation varies while interface stays the same

Second view: share code, factor code into parent class
Polymorphism/late/implement 
Function actually called determined when program runs,

Who is Alan Perlis?
It is easier to write an incorrect program than to understand a correct one.

- Simplicity does not precede correctness, but follows it.
- If you have a procedure with ten parameters you probably missed some.
- If a listener nods his head when you're explaining your program, wake him up.
- Programming is an unnatural act.

Review/Preview: Anagrams/Jumbles
Brute-force approach to finding anagrams/solving Jumbles

- Brute-force often thought of as "Jack of all trades, master of none".
- What if there's nothing better?
- What if the better way requires too much thought?
- What's the problem here?
- Is there a better method?

- nelir, neri, neir, nir, ire
- nel, nei, nei, nei, nei, nei
- nleir, nleri, nlier, nler, ler, ler
- nilir, nleri, nleri, nler, ler, ler

nelir, nelri, nelri, nelri, nelri
nlire, nlrei, nlrei, nlrei, nlrei

nel, nei, nei, nei, nei
nleir, nleri, nleri, nleri, nleri

nel, nei, nei, nei, nei
nleir, nleri, nleri, nleri, nleri

nel, nei, nei, nei, nei
nleir, nleri, nleri, nleri, nleri

nel, nei, nei, nei, nei
nleir, nleri, nleri, nleri, nleri