Welcome to Computer Science (CS311 Yoshii)

What is Computer Science?

It is about solving the world's problems using a computer (usually by creating software).

It leads to brand new technologies!

What problems? Look around you – there are many!
See https://sdgs.un.org/goals for the 17 goals of the world!

- **Crime**
- **Global warming**
- **Medical**
- **Education**
- **Transportation**
- **Business (CIS)**
- **Etc. etc.**

All CS majors should be thinking of problems they want to solve.

You may even want to minor in a subject you are interested.

What is Programming then?

Teach the computer how to solve the problem by

stating the algorithm (i.e. detailed instructions) in a special language.

This special language is easy for humans to read and write (e.g. C++, Java, C#, Python..)

But the computer itself does not understand it.

So, a program called a Compiler will translate it.

A compiler is machine dependent.

The translated code is machine dependent.

Stating the Algorithm isn't Easy.

The computer can only do very simple tasks such as

- **Put a number in a box**
- **Compare contents of boxes**
- **Do arithmetic on box contents**
- **Show the result to the user**

Fitting your solution to a difficult problem into this limited repertory

is very difficult. But that is the main task of a software engineer.

It is like using LEGO blocks to build something complex.

That and the fact that you need to understand the field of the problem is why you get paid a lot!

Checkout the Success Stories page.

Getting A Job In CS Is Not Easy

First of all, your CS GPA has to be high.

You have to have personal projects listed in your resume to show how much you love CS.

And pass a series of interviews (tests).

Not everyone with a CS degree will be able to get a job in CS.

Test and Debug and Update

You will have to test your program many, many times and remove errors (debug).

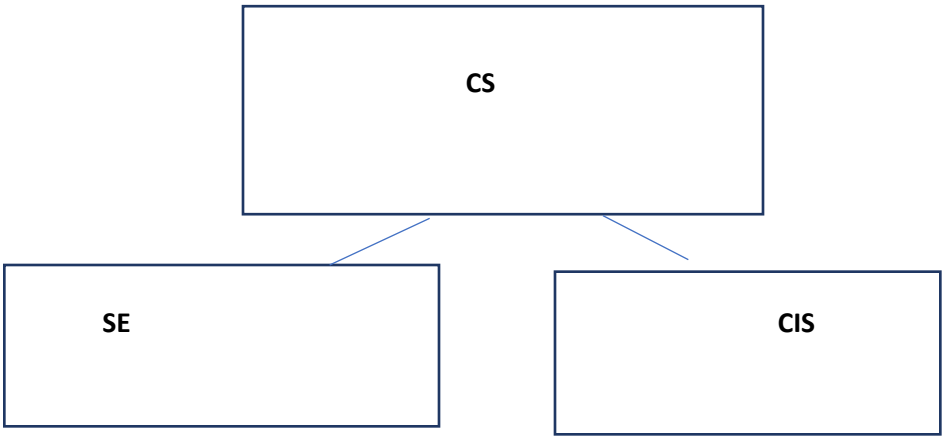
Making it completely bug free is nearly impossible!!

But we must do our best.

Please read help-cpp and use the testing and debugging techniques explained there.

This leads to a sub-field called Software Engineering.

- **How do we create software quickly by a lot of different groups working together?**
- **How do we make sure the software is easy to update (maintain)?**
  80% of software cost is due to maintenance.
- **Can we make sure the software parts are reusable?**
- **How can we test and debug easily?**



What You will be Learning as a CS major

Fundamentals of Programming (in C++) (CS111/211)

- **Simple algorithms**
- **Typing a program**
- **Compiling it**
- **Testing it**
- **Debug it**

What is inside the computer? (CS231/331)

- **The parts (memory, CPU, etc)**
- **Instruction set/the computer's language**
- **Binary arithmetic (0s and 1s)**

Algorithm and Data Structure Choices (CS311)

- **Well-known algorithms and data structures**
- **How do you choose an algorithm?**
- **How do you choose a structure for storing data?**
- **Mathematical analysis (Discrete math)**

Features of Programming Languages (CS351)

- **Well-known features of programming languages**
- **Advantages and disadvantages**
- **Compiled vs Interpreted**
- **How does a Compiler work? (Grammar analysis)**

Operating Systems (CS433)

- **OS tasks**
- **Well-known OS features**
- **Advantages and disadvantages**

Networks (CS436)

- **How do devices communicate with each other?**
- **Security? Speed? Reliability?**

Software Engineering Methodologies (CS441 and SE classes)

Elective Sub-Fields: (all becoming essential these days)

Embedded Systems and IOT

Artificial Intelligence (Machine Learning, Text Mining, NLP, Self-Driving cars, ...)

- **Probability and Calc classes are important**

VR, Graphics, Games (engines)

- **Linear Algebra class is important**

Bioinformatics (DNA analysis)

Mobile computing technology

Cloud computing technology

Cybersecurity (Detection, ethical hacking, encryption, restoration, …)

Checkout the Faculty and Club Pages now.