

Writing to design and communicate in computer science

Dr. Chris Bourke
Dept. of Computer Science & Engineering

Dr. Meredith Steck & Charlotte Kupsh
The Writing Center

Figure 3: Need vs. Proficiency of Career Readiness Competencies, by Percent of Respondents

COMPETENCIES	CONSIDERED ESSENTIAL*	RATED PROFICIENT**
Teamwork/Collaboration		
Digital Technology		
Critical Thinking/Problem Solving		
Professionalism/Work Ethic		
Oral/Written Communications		
Leadership		
Global/Multi-Cultural Fluency		
Career Management		

Source: Job Outlook 2018, National Association of Colleges and Employers.

Figure 3: Need vs. Proficiency of Career Readiness Competencies, by Percent of Respondents

COMPETENCIES	CONSIDERED ESSENTIAL*	RATED PROFICIENT**
Teamwork/Collaboration	97.5%	
Digital Technology	64.2%	
Critical Thinking/Problem Solving	99.2%	
Professionalism/Work Ethic	100%	
Oral/Written Communications	95.9%	
Leadership	68.6%	
Global/Multi-Cultural Fluency	31.1%	
Career Management	47.1%	

Source: Job Outlook 2018, National Association of Colleges and Employers.

Figure 3: Need vs. Proficiency of Career Readiness Competencies, by Percent of Respondents

COMPETENCIES	CONSIDERED ESSENTIAL*	RATED PROFICIENT**
Teamwork/Collaboration	97.5%	77.0%
Digital Technology	64.2%	65.8%
Critical Thinking/Problem Solving	99.2%	55.8%
Professionalism/Work Ethic	100%	42.5%
Oral/Written Communications	95.9%	41.6%
Leadership	68.6%	33.0%
Global/Multi-Cultural Fluency	31.1%	20.7%
Career Management	47.1%	17.3%

Source: Job Outlook 2018, National Association of Colleges and Employers.

Agenda

1. Freewrite: The parallels between coding and writing (10 min)
2. Reverse outlines: how they can help you learn to code (20 min)
3. Analyzing design documents: how they communicate your code (20 min)

Agenda

1. Freewrite: The parallels between coding and writing (10 min)
2. Reverse outlines: how they can help you learn to code (20 min)
3. Analyzing design documents: how they communicate your code (20 min)

Freewrite: The parallels between coding and writing

In "Writing Good Code is Like Writing a Novel", the author posits that the process of writing code and the process of writing a novel are similar. In one section, the author focuses on DRAFTS.

Think about how you draft an essay or any piece of writing.

- What are the steps you take to get from an idea to a final draft?
- How are these steps similar to your coding process?

Take 4 min to freewrite!

Reverse Outlines

How do I make a reverse outline?

1. Start with a complete draft
2. Summarize the function of each paragraph in one sentence
3. Number your paragraphs

Reverse Outlines

How do I make a reverse outline?

1. Start with a complete draft
2. Summarize the function of each paragraph in one sentence
3. Number your paragraphs

To the Members of the Hiring Committee:

I am writing with great enthusiasm to apply for the position of Learning Commons Supervisor at Cuyahoga Community College. I am currently the Associate Director of the Writing Center at the University of Nebraska-Lincoln.

I have 7+ years of experience providing tutoring and administrative support in a learning commons environment. As the Associate Director of the Writing Center, I am currently responsible for supervising a staff of 20 undergraduate and graduate tutors who meet individually with students, staff and faculty across the disciplines to conference about their writing. My responsibilities include....

Reverse Outlines

How do I make a reverse outline?

1. Start with a complete draft
2. Summarize the function of each paragraph in one sentence
3. Number your paragraphs

To the Members of the Hiring Committee:

I am writing with great enthusiasm to apply for the position of Learning Commons Supervisor at Cuyahoga Community College. I am currently the Associate Director of the Writing Center at the University of Nebraska-Lincoln.

I have 7+ years of experience providing tutoring and administrative support in a learning commons environment. As the Associate Director of the Writing Center, I am currently responsible for supervising a staff of 20 undergraduate and graduate tutors who meet individually with students, staff and faculty across the disciplines to conference about their writing. My responsibilities include....

1. Introducing myself and my purpose to the reader

Reverse Outlines

How do I make a reverse outline?

1. Start with a complete draft
2. Summarize the function of each paragraph in one sentence
3. Number your paragraphs

To the Members of the Hiring Committee:

I am writing with great enthusiasm to apply for the position of Learning Commons Supervisor at Cuyahoga Community College. I am currently the Associate Director of the Writing Center at the University of Nebraska-Lincoln.

I have 7+ years of experience providing tutoring and administrative support in a learning commons environment. As the Associate Director of the Writing Center, I am currently responsible for supervising a staff of 20 undergraduate and graduate tutors who meet individually with students, staff and faculty across the disciplines to conference about their writing. My responsibilities include....

1. Introducing myself and my purpose to the reader

2. Explaining why I am qualified for the job

Now, let's try reverse outlining code!

Exercise: Suffix Arrays

Compute and sort the **suffixes** of a given word.

Input: **science**

Suffixes:	Sorted:
science	ce
cience	cience
ience	e
ence	ence
nce	ience
ce	nce
e	science

Reverse outline this code:

```
List<String> suffixes = new ArrayList<>();
for(int i=0; i<args[0].length(); i++) {
    suffixes.add(args[0].substring(i));
}

Collections.sort(suffixes);

for(String s : suffixes) {
    System.out.println(s);
}
```

Reverse outline this code:

```
List<String> suffixes = new ArrayList<>();
for(int i=0; i<args[0].length(); i++) {
    suffixes.add(args[0].substring(i));
}

Collections.sort(suffixes);

for(String s : suffixes) {
    System.out.println(s);
}
```

Create a collection of suffixes from the first
 command line argument
 i. Initialize the collection
 ii. Iterate over the string to create
 suffixes, adding them to the collection
 Sort the collection
 Print the collection to the standard output

```
int i, j, k = 0;

//Scanner scan = new Scanner(System.in);
//s = scan.next();
String s = args[0];
//System.out.println(s);
String temp; temp;

String tempArray[] = null;
tempArray = s.split("");
String suffixes[] = new String[tempArray.length];
for(i=0; i<tempArray.length; i++) {
    suffixes[i] = "";
}

int n = s.split("").length;
for(i=0; i<n; i++) {
    for(j=i; j<n; j++) {
        suffixes[i] = suffixes[i] + suffixes[j];
    }
}

for(i=0; i<suffixes.length; i++) {
    int minIndex = i;
    for(j=i; j<suffixes.length; j++) {
        if(suffixes[i].compareTo(suffixes[minIndex]) > 0) {
            minIndex = j;
        }
    }
    temp = suffixes[minIndex];
    suffixes[minIndex] = suffixes[i];
    suffixes[i] = temp;
}

String result = "";
for(i=0; i<suffixes.length; i++) {
    result = result + suffixes[i];
    result = result + "\n";
}

System.out.println(result);
```

```
int i, j, k = 0;

//Scanner scan = new Scanner(System.in);
//s = scan.next();
String s = args[0];
//System.out.println(s);
String temp; temp;

String tempArray[] = null;
tempArray = s.split("");
String suffixes[] = new String[tempArray.length];
for(i=0; i<tempArray.length; i++) {
    suffixes[i] = "";
}

int n = s.split("").length;
for(i=0; i<n; i++) {
    for(j=i; j<n; j++) {
        suffixes[i] = suffixes[i] + suffixes[j];
    }
}

for(i=0; i<suffixes.length; i++) {
    int minIndex = i;
    for(j=i; j<suffixes.length; j++) {
        if(suffixes[i].compareTo(suffixes[minIndex]) > 0) {
            minIndex = j;
        }
    }
    temp = suffixes[minIndex];
    suffixes[minIndex] = suffixes[i];
    suffixes[i] = temp;
}

String result = "";
for(i=0; i<suffixes.length; i++) {
    result = result + suffixes[i];
    result = result + "\n";
}

System.out.println(result);
```

AAAAAAAAAAHHH
HHHHHH!!!!

Design documents: Writing to communicate

What are you writing? Who is it for? Why is it important?

Design documents: Writing to communicate

What are you writing? Who is it for? Why is it important?

Design Document

Due: Apr 18 by 11:59pm Points: 100

- **Overview & Requirements**
 - **Index**
 - Word Template
 - [designDocument-template.tex](#)
 - [designDocument-template.pdf](#)
 - [bibliography.bib](#)
 - LaTeX Template
 - [designDocument-template.tex](#)
 - [designDocument-template.pdf](#)
 - [bibliography.bib](#)
 - [IEEE1016-2007](#) (Design Document specification)
 - Examples
 - [Professional Example](#)
 - [Student Example 1](#)
 - [Student Example 2](#)
 - [Student Example 3](#)

Design documents: Writing to communicate

What are you writing? Who is it for? Why is it important?

Design Document

Due: Apr 18 by 11:59pm Points: 100

- **Overview & Requirements**
 - **Index**
 - Word Template
 - [designDocument-template.tex](#)
 - [designDocument-template.pdf](#)
 - [bibliography.bib](#)
 - LaTeX Template
 - [designDocument-template.tex](#)
 - [designDocument-template.pdf](#)
 - [bibliography.bib](#)
 - [IEEE1016-2007](#) (Design Document specification)
 - Examples
 - [Professional Example](#)
 - [Student Example 1](#)
 - [Student Example 2](#)
 - [Student Example 3](#)

The Design Document should be clear and comprehensive enough that a reasonably technically competent person could reproduce your system design (though not necessarily every detail) with identical functionality without access to your actual code base. Our evaluation will be based on how well your Design Document achieves this principle. In addition, we will also evaluate your technical writing skills by evaluating the following items. Note that this is not an exhaustive list.

Design documents: Writing to communicate

What are you writing? Who is it for? Why is it important?

Design Document

Due: Apr 18 by 11:59pm Points: 100

- **Overview & Requirements**
 - **Index**
 - Word Template
 - [designDocument-template.tex](#)
 - [designDocument-template.pdf](#)
 - [bibliography.bib](#)
 - LaTeX Template
 - [designDocument-template.tex](#)
 - [designDocument-template.pdf](#)
 - [bibliography.bib](#)
 - [IEEE1016-2007](#) (Design Document specification)
 - Examples
 - [Professional Example](#)
 - [Student Example 1](#)
 - [Student Example 2](#)
 - [Student Example 3](#)

The Design Document should be clear and comprehensive enough that a reasonably technically competent person could reproduce your system design (though not necessarily every detail) with identical functionality without access to your actual code base. Our evaluation will be based on how well your Design Document achieves this principle. In addition, we will also evaluate your technical writing skills by evaluating the following items. Note that this is not an exhaustive list.

2 Design Overview

2.1 Approach

This document is created and extended in multiple phases over the course of the project -

- **Requirements Phase** - During the Requirements Phase, the initial version of this document is created, describing the candidate architecture to be validated in the System Design Phase.
- **System Design Phase** - During the System Design Phase, the Evolutionary Prototype is created and this document is finalized by establishing a sound architectural foundation for the Construction Phase.
- **Construction Phase** - During the Construction Phase, this document is not expected to change radically; it is mainly updated to reflect changes in any interface definitions.
- **Transition / Training Phase** - During the Transition/Training Phase, no further additions or modifications are made to this document.

2.2 Architectural Goals and Constraints

The overall architecture goals of the system is to provide a highly available and scalable online screening tool for users of the State of Indiana, to understand what programs or services are available and to determine if they are potentially eligible for those services.

The Online Screening Tool can be used in two ways -

- To answer questions related to Food Stamps, TANF and Health Coverage services and determine potential eligibility for any of these programs
- To generate reports based on screening program data for stakeholders

A key Architectural goal is to leverage industry best practices for designing and developing a scalable, enterprise-wide (EIS) application. To meet this goal, the design of the Online Screening Tool will be based on core EIS patterns as well as the industry standard development guidelines for building the Online Screening Tool.

2.1 Approach

This document is created and extended in multiple phases over the course of the project -

- **Requirements Phase** - During the Requirements Phase, the initial version of this document is created, describing the candidate architecture to be validated in the System Design Phase.
- **System Design Phase** - During the System Design phase, the Evolutionary Prototype is created and this document is finalized by establishing a sound architectural foundation for the Construction Phase.
- **Construction Phase** - During the Construction Phase, this document is not expected to change radically; it is mainly updated to reflect changes in any interface definitions.
- **Transition / Training Phase** - During the Transition/Training Phase, no further additions or modifications are made to this document.

6.1 Data Model

Data Model is a method for describing data structures and a set of operations used to manipulate and validate that data. Data Model for the Online Screening application is as shown below -

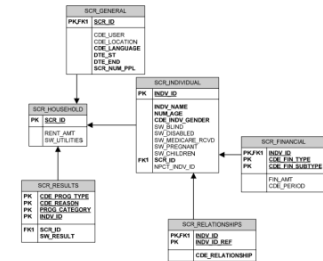


Fig 9: Data Model for Screening tool

UNL Writing Center

<https://www.unl.edu/writing> ; writingcenter@unl.edu

We offer one-on-one sessions with undergraduate and graduate peer consultants for writing in any discipline or any stage of the writing process, whether you are brainstorming, organizing your ideas, polishing a final draft, or anything in between.



UNL Writing Center

<https://www.unl.edu/writing> ; writingcenter@unl.edu

- **102 Andrews Hall**
- Evening hours from 5-7 pm in the Adele Learning Commons
- Satellite locations in the Gaughan Multicultural Center, Nebraska Hall, and the East Campus Union
- Visit unl.mywconline.com to schedule a 25 or 50-minute appointment online
- **What to bring when you come?**



The Engineering Satellite!



Olsson Room in the Engineering Library in Nebraska Hall
 Monday, Tuesday, Thursday: 7:30-9pm
 Tuesday: 12:30-2:30pm