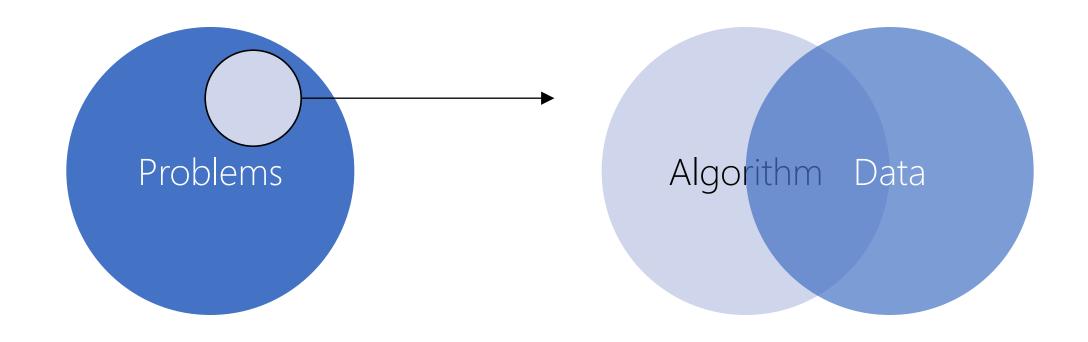Problems

Computable
Theory of Automata
Theory of Computation
Theory of Computer Science
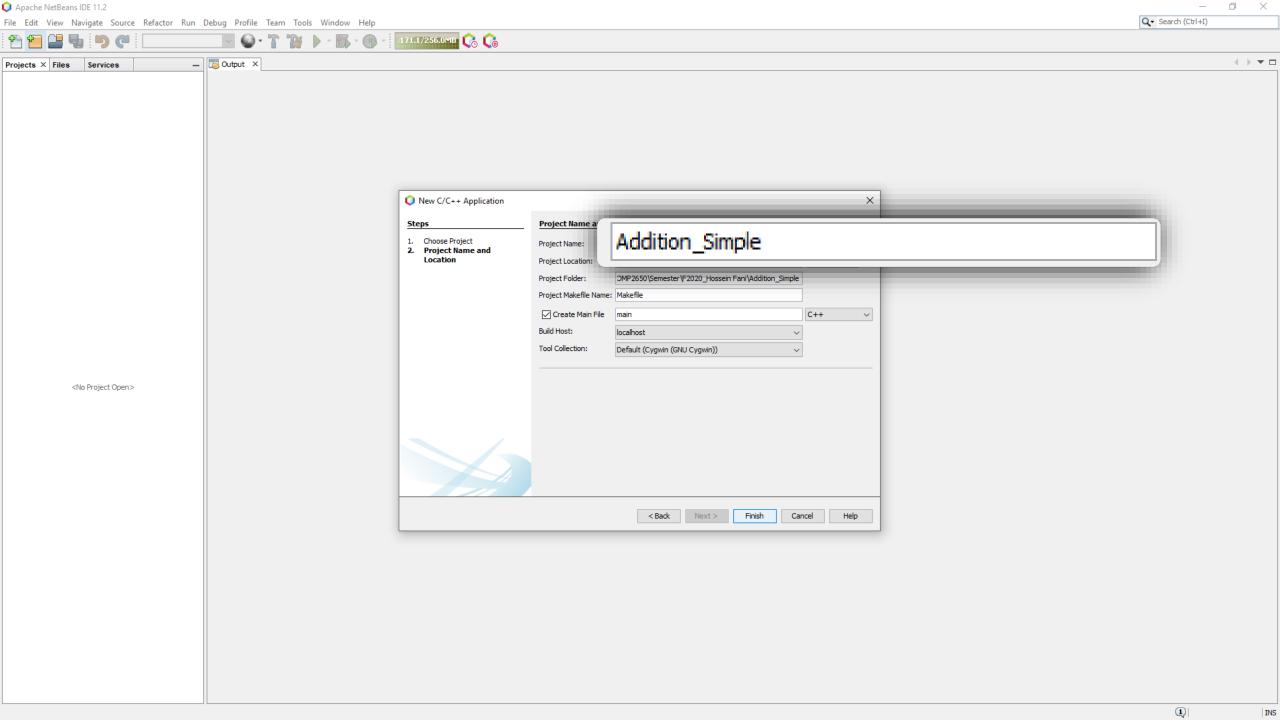
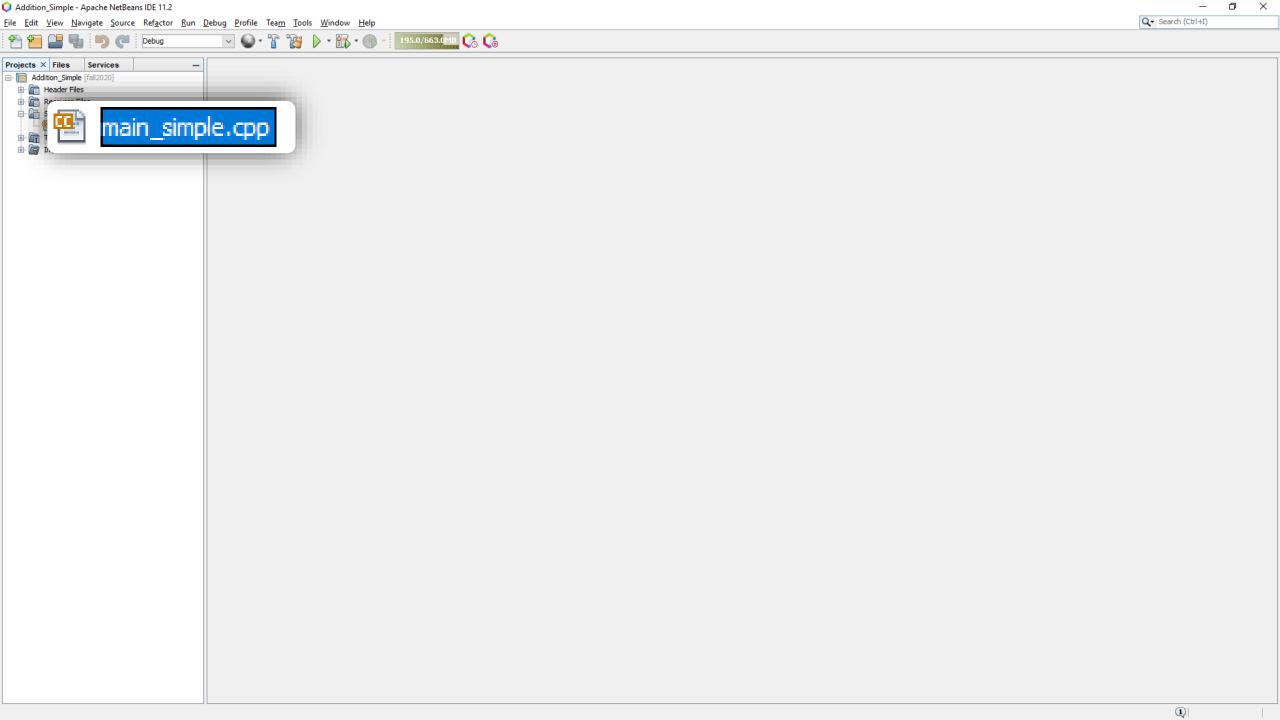Problems

Problems

Algorithm   Data

Algorithm Design
Algorithm Analysis
Artificial Intelligence (AI)
Machine Learning
Data Mining

Algo Data
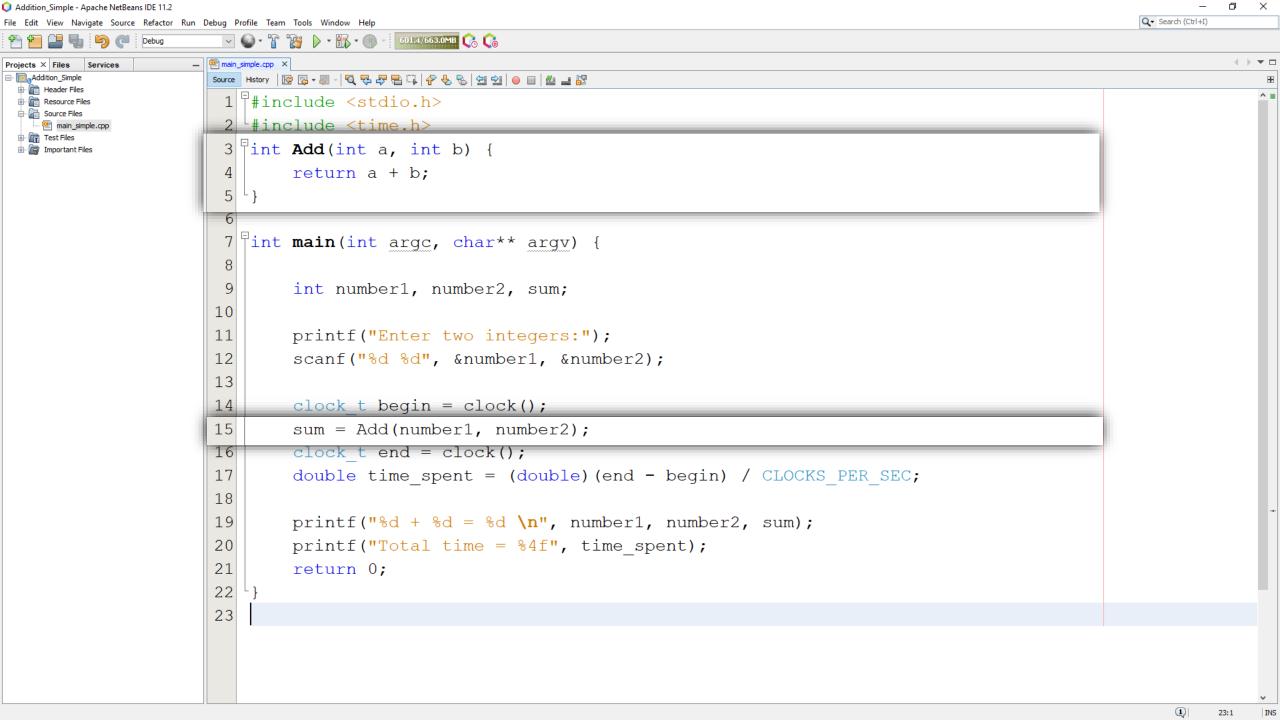
Data Structure (Memory)
File Structure (File Systems)
Database Management Sys.
Data Warehouse
Big Data
Cloud

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

166.0/256.0MB

Projects Files Services

Output

<No Project Open>

New Project

Steps

1. **Choose Project**
2. ...

Choose Project

Filter:

Categories:

- Java with Maven
- Java with Gradle
- Java with Ant
- HTML5/JavaScript
- PHP
- C/C++
- Samples

Projects:

- C/C++ Project with Existing Sources
- C/C++ Project from Binary Files
- **C/C++ Application**
- C/C++ Qt Application
- C/C++ Qt Dynamic Library
- C/C++ Qt Static Library

Description:

Creates a new application project. It uses an IDE-generated makefile to build your project.

< Back   Next >   Finish   Cancel   Help

INS

File   Edit   View   Navigate   Source   Refactor   Run   Debug   Profile   Team   Tools   Window   Help

Search (Ctrl+I)

171.1/256.0MB

Output ×

<No Project Open>

New C/C++ Application

Steps

1. Choose Project
2. **Project Name and Location**

Project Name a

Project Name:

Addition_Simple

Project Location:

Project Folder: OMP2650\Semester\F2020_Hossein Fani\Addition_Simple

Project Makefile Name: Makefile

☑ Create Main File   main   C++

Build Host: localhost

Tool Collection: Default (Cygwin (GNU Cygwin))

< Back   Next >   Finish   Cancel   Help

INS

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Debug

195.0/663.0MB

Projects × Files Services

Addition_Simple [fall2020]
  Header Files
  Resource Files

main_simple.cpp

File   Edit   View   Navigate   Source   Refactor   Run   Debug   Profile   Team   Tools   Window   Help

main_simple.cpp

Source   History

```c
#include <stdio.h>
#include <time.h>

int Add(int a, int b) {
    return a + b;
}

int main(int argc, char** argv) {

    int number1, number2, sum;

    printf("Enter two integers:");
    scanf("%d %d", &number1, &number2);

    clock_t begin = clock();
    sum = Add(number1, number2);
    clock_t end = clock();
    double time_spent = (double)(end - begin) / CLOCKS_PER_SEC;

    printf("%d + %d = %d \n", number1, number2, sum);
    printf("Total time = %4f", time_spent);
    return 0;
}
```

```
BUILD SUCCESSFUL (total time: 756ms)
Enter two integers:
12
35
12 + 35 = 47
Total time = 0.000000
```

```
BUILD SUCCESSFUL (total time: 756ms)
Enter two integers:
12
-35
12 + -35 = -23
Total time = 0.000000
```

```
BUILD SUCCESSFUL (total time: 775ms)
Enter two integers:
999999
1
999999 + 1 = 1000000
```

```
BUILD SUCCESSFUL (total time: 775ms)
Enter two integers:
999999999999999
1
```

Enter two integers:
999999999999999
1
-1530494977 + 1 = -1530494976
Total time = 0.000000

```
BUILD SUCCESSFUL (total time: 775ms)
Enter two integers:
999999999999999
1
-1530494977 + 1 = -1530494976
Total time = 0.000000
```

```
File   Edit   View   Navigate   Source   Refactor   Run   Debug   Profile   Team   Tools   Window   Help
```

```
Debug
```

```
main_simple.cpp
```

Source   History

```cpp
 1  #include <stdio.h>
 2  #include <limits.h>
 3  #include <time.h>
 4  int Add(int a, int b) {
 5      return a + b;
 6  }
 7
 8  int main(int argc, char** argv) {
 9
10      int number1, number2, sum;
11
12      printf("Enter two integers between %d and %d: \n", INT_MIN, INT_MAX);
13      scanf("%d %d", &number1, &number2);
14
15      clock_t begin = clock();
16      sum = Add(number1, number2);
17      clock_t end = clock();
18      double time_spent = (double)(end - begin) / CLOCKS_PER_SEC;
19
20      printf("%d + %d = %d \n", number1, number2, sum);
21      printf("Total time = %4f", time_spent);
22      return 0;
23  }
24
```

main_simple.cpp saved.   24:1   INS

```
BUILD SUCCESSFUL (total time: 715ms)
Enter two integers between -2147483648 and 2147483647:
12
15
12 + 15 = 27
Total time = 0.000000
```

BUILD SUCCESSFUL (total time: 715ms)
Enter two integers between -2147483648 and 2147483647:
2147483647
1

BUILD SUCCESSFUL (total time: 715ms)
Enter two integers between -2147483648 and 2147483647:
2147483647
1
2147483647 + 1 = -2147483648
Total time = 0.000000

```
BUILD SUCCESSFUL (total time: 805ms)
Enter two integers between -2147483648 and 2147483647:
2147483647
2
2147483647 + 2 = -2147483647
Total time = 0.000000
```

```
Enter two integers:
2147483647
1
2147483647 + 1 = 2147483648
```

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

main_moderate.cpp

Source History

```cpp
//Chitra Nayal, Mithun Kumar, Shivam Kushik, 04-06-2019, Sum of two large numbers
//Retrieved from https://www.geeksforgeeks.org/sum-two-large-numbers/
#include <stdio.h>
#include<bits/stdc++.h>
using namespace std;

string Add(string a, string b) {
    // Before proceeding further, make sure length of str2 is larger.
    if (a.length() > b.length())
            swap(a, b);

    string result = "";

    // Calculate length of both string
    int n1 = a.length();
    int n2 = b.length();

    // Reverse both of strings
    reverse(a.begin(), a.end());
    reverse(b.begin(), b.end());

    int carry = 0;
    for (int i = 0; i < n1; i++) {
        // Do school mathematics, compute sum of current digits and carry
        int sum = ((a[i] - '0') + (b[i] - '0') + carry);
        result.push_back(sum % 10 + '0');
        // Calculate carry for next step
        carry = sum / 10;
    }

    // Add remaining digits of larger number
    for (int i = n1; i < n2; i++) {
        int sum = ((b[i] - '0') + carry);
        result.push_back(sum % 10 + '0');
        carry = sum / 10;
    }

    // Add remaining carry
    if (carry)
        result.push_back(carry + '0');

    // reverse resultant string
    reverse(result.begin(), result.end());

    return result;
}

int main(int argc, char** argv) {...17 lines }

```

```
//Chitra Naval, Mithun Kumar, Shivam Kushik, 04-06-2019, Sum of two large numbers
//Retrieved from https://www.geeksforgeeks.org/sum-two-large-numbers/
```

```cpp
    int carry = 0;
    for (int i = 0; i < n1; i++) {
        // Do school mathematics, compute sum of current digits and carry
        int sum = ((a[i] - '0') + (b[i] - '0') + carry);
        result.push_back(sum % 10 + '0');
        // Calculate carry for next step
        carry = sum / 10;
    }
```

File   Edit   View   Navigate   Source   Refactor   Run   Debug   Profile   Team   Tools   Window   Help

Debug

353.2/371.0MB

main_moderate.cpp ×

Source   History

```cpp
1   #include <stdio.h>
2   #include<bits/stdc++.h>
3   using namespace std;
4
5   string Add(string a, string b) {...40 lines }
45
46  int main(int argc, char** argv) {
47
48      string number1, number2, sum;
49
50      cout<<"Enter two integers: \n";
51      cin >> number1 >> number2;
52
53      clock_t begin = clock();
54      sum = Add(number1, number2);
55      clock_t end = clock();
56      double time_spent = (double)(end - begin) / CLOCKS_PER_SEC;
57
58      cout<<number1<<" + "<<number2<<" = "<<sum<<"\n";
59      cout<<"Total time = "<<fixed<<setprecision(6)<<time_spent;
60
61      return 0;
62  }
63
```

main_moderate.cpp saved.

63:1   INS

BUILD SUCCESSFUL (total time: 715ms)
Enter two integers:
2147483647
1
2147483647 + 1 = 2147483648
Total time = 0.000000

Enter two integers:
99999999999999999999999999999999999999
1
99999999999999999999999999999999999999 + 1 = 100000000000000000000000000000000000000
Total time = 0.000000

BUILD SUCCESSFUL (total time: 715ms)
Enter two integers:
999...99999 => 50,000 digits
999...99999 => 50,000 digits
999...99999 + 999...99999 = 1999 ... 9998 => 50,001 digits
Total time = 0.000000

```
BUILD SUCCESSFUL (total time: 715ms)
Enter two integers:
999...99999 => 50,000 digits
999...99999 => 50,000 digits
How many iterations:
?
Total time = ?
```

Enter two integers:
999...99999 => 50,000 digits
999...99999 => 50,000 digits
How many iterations:
1
Total time = 0.000000

Enter two integers:
999...99999 => 50,000 digits
999...99999 => 50,000 digits
How many iterations:
100
Total time = ?

Problems

Algorithm Data

Machine

Algorithm Design
Algorithm Analysis
Artificial Intelligence (AI)
Machine Learning
Data Mining

Data Structure
File Structure
Database Management Sys.
Data Warehouse
Big Data
Cloud

Algo

Data

Machine

Digital Design (Logic Circuits)
Computer Architecture
Microprocessor Programming

Algorithm Design
Algorithm Analysis
Artificial Intelligence (AI)
Machine Learning
Data Mining

Algo   Data

Machine

Data Structure
File Structure
Database Management Sys.
Data Warehouse
Big Data
Cloud

Digital Design (Logic Circuits)
Computer Architecture
Microprocessor Programming

DIGITAL DESIGN

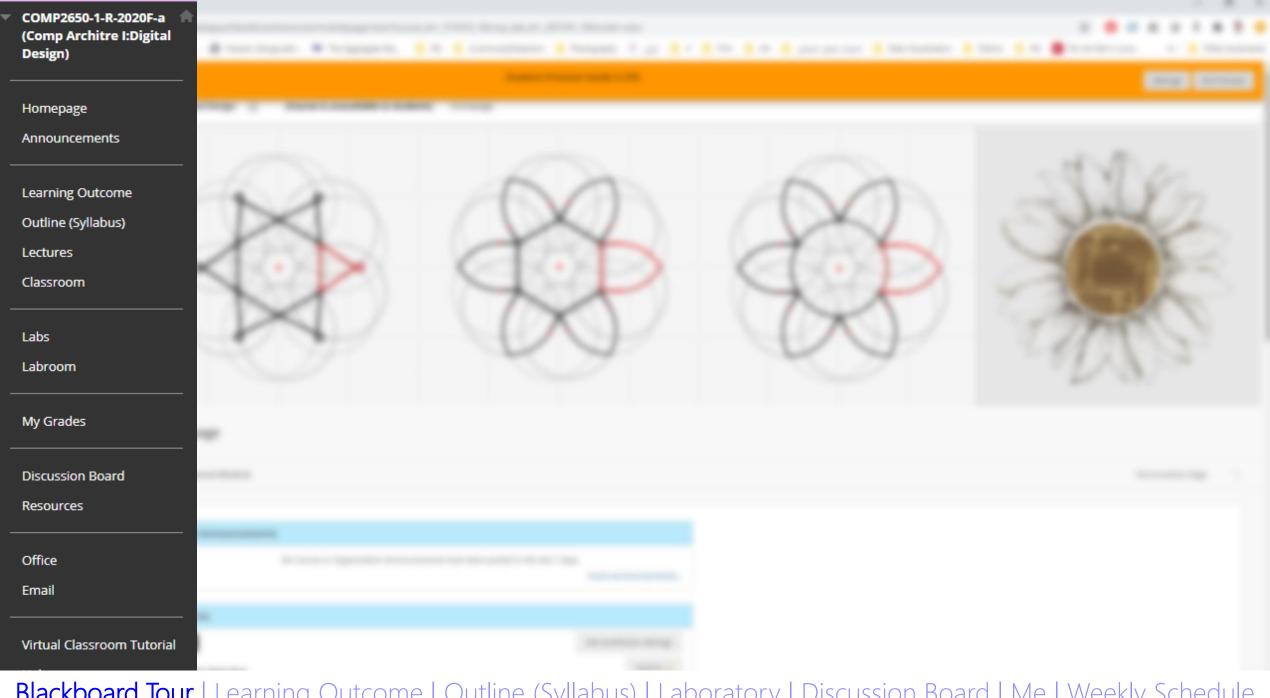aka. Logic Circuits

Fall 2020

DIGITAL DESIGN
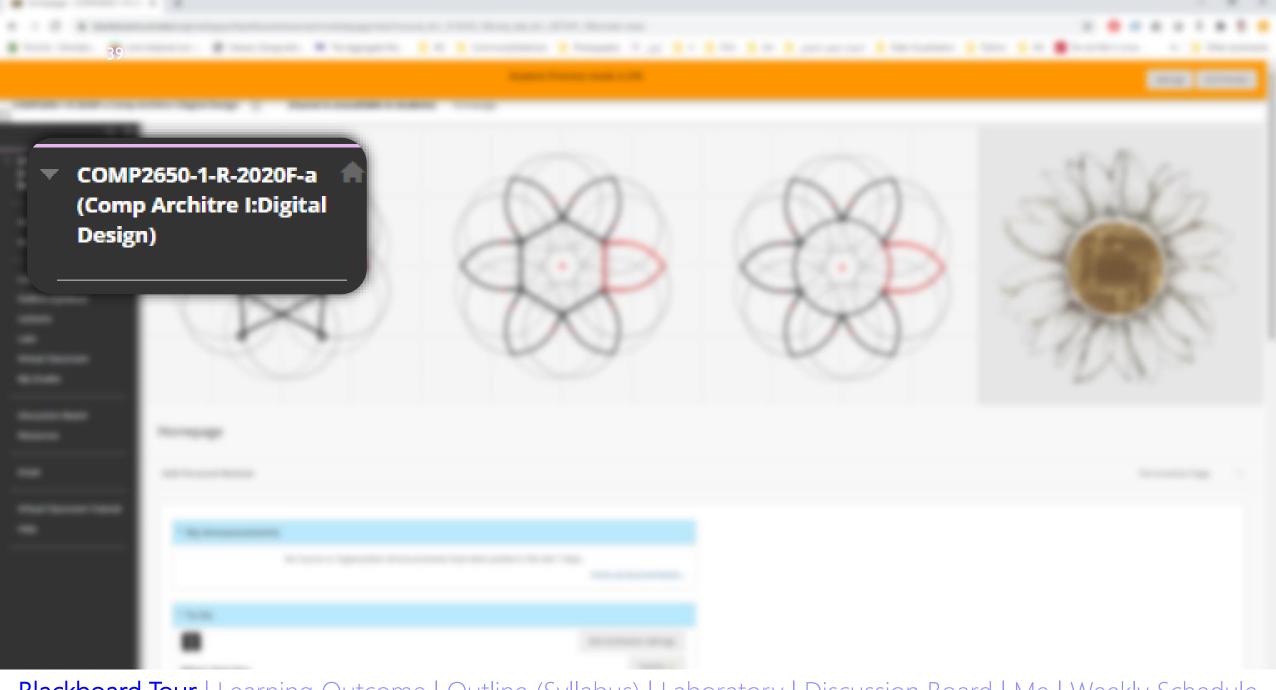
aka. Logic Circuits

|---> blackboard.uwindsor.ca
  |---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design

BLACKBOARD TOUR

COMP2650-1-R-2020F-a (Comp Architre I:Digital Design)

Homepage

Announcements

Learning Outcome

Outline (Syllabus)

Lectures

Classroom

Labs

Labroom

My Grades

Discussion Board

Resources

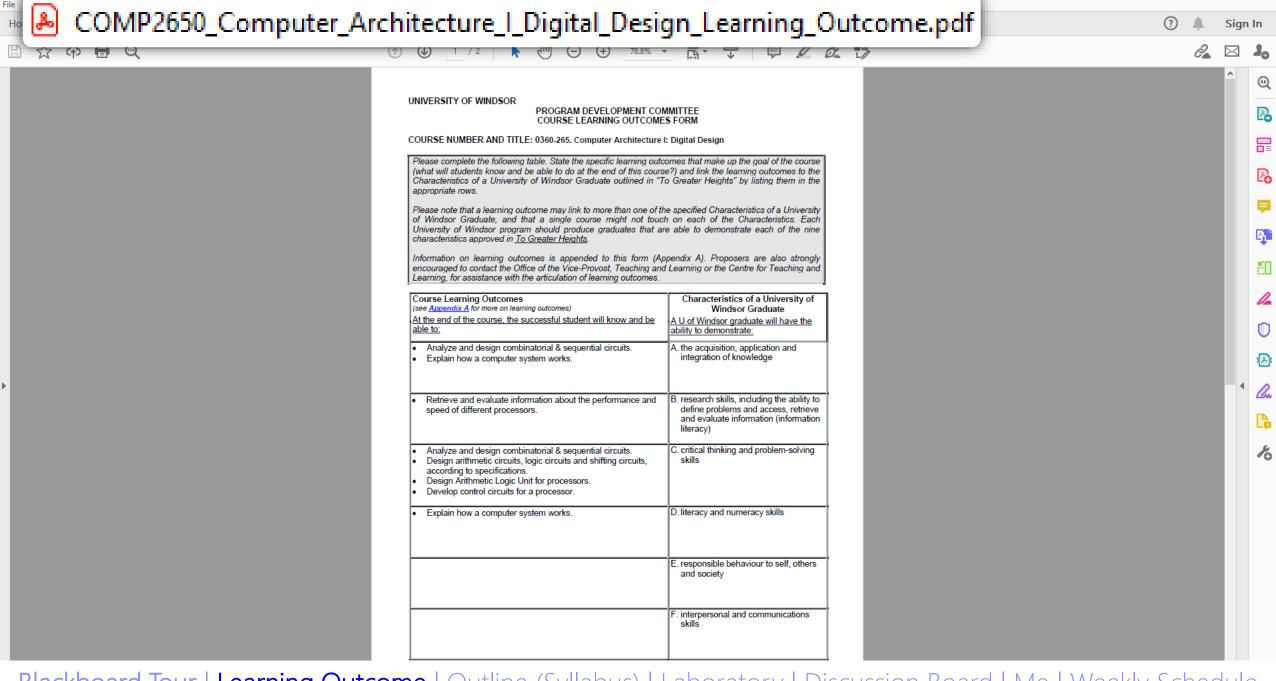Office

Email

Virtual Classroom Tutorial
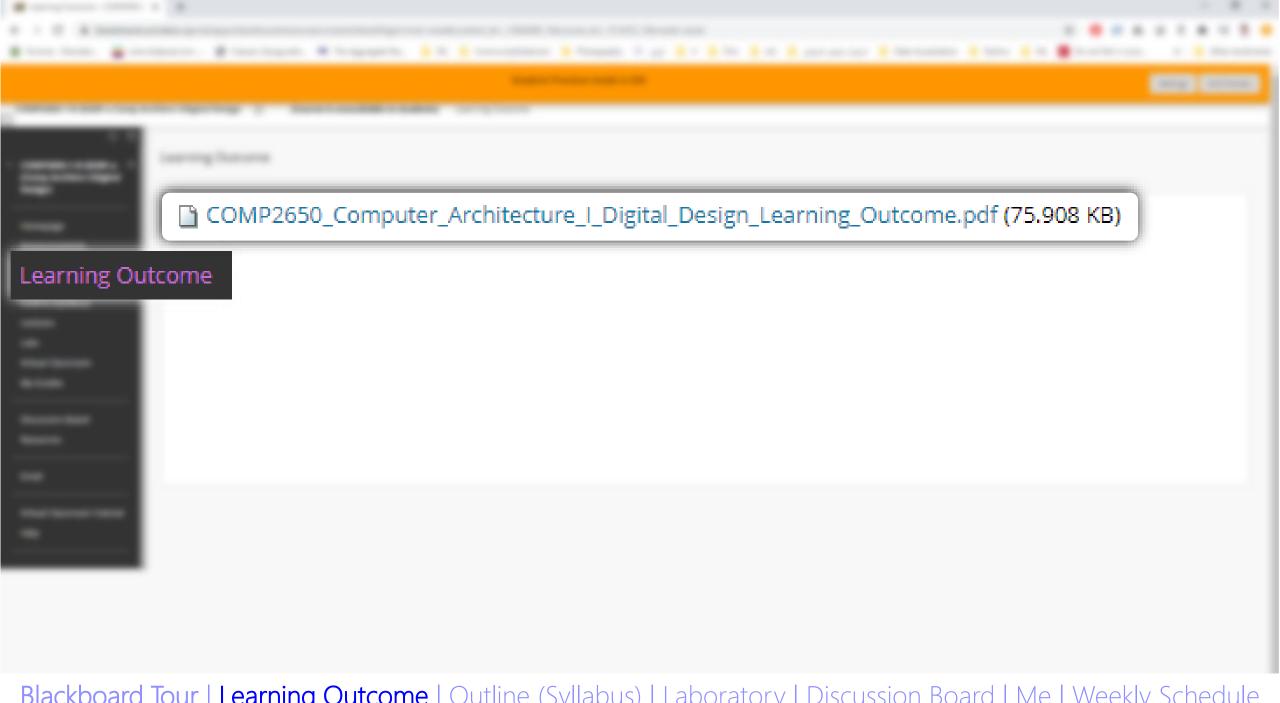
LEARNING OUTCOME

aka. Learning Objectives

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
   |---> Learning Outcome
     |---> COMP2650_Computer_Architecture_I_Digital_Design_Learning_Outcome.pdf

📄 COMP2650_Computer_Architecture_I_Digital_Design_Learning_Outcome.pdf (75.908 KB)

Learning Outcome

UNIVERSITY OF WINDSOR

PROGRAM DEVELOPMENT COMMITTEE
COURSE LEARNING OUTCOMES FORM

COURSE NUMBER AND TITLE: 0360-265. Computer Architecture I: Digital Design

Please complete the following table. State the specific learning outcomes that make up the goal of the course (what will students know and be able to do at the end of this course?) and link the learning outcomes to the Characteristics of a University of Windsor Graduate outlined in "To Greater Heights" by listing them in the appropriate rows.

Please note that a learning outcome may link to more than one of the specified Characteristics of a University of Windsor Graduate, and that a single course might not touch on each of the Characteristics. Each University of Windsor program should produce graduates that are able to demonstrate each of the nine characteristics approved in <u>To Greater Heights</u>.

Information on learning outcomes is appended to this form (Appendix A). Proposers are also strongly encouraged to contact the Office of the Vice-Provost, Teaching and Learning or the Centre for Teaching and Learning, for assistance with the articulation of learning outcomes.

Course Learning Outcomes (see *Appendix A* for more on learning outcomes) <u>At the end of the course, the successful student will know and be able to:</u>	Characteristics of a University of Windsor Graduate <u>A U of Windsor graduate will have the ability to demonstrate:</u>
• Analyze and design combinatorial & sequential circuits. • Explain how a computer system works.	A. the acquisition, application and integration of knowledge
• Retrieve and evaluate information about the performance and speed of different processors.	B. research skills, including the ability to define problems and access, retrieve and evaluate information (information literacy)
• Analyze and design combinatorial & sequential circuits. • Design arithmetic circuits, logic circuits and shifting circuits, according to specifications. • Design Arithmetic Logic Unit for processors. • Develop control circuits for a processor.	C. critical thinking and problem-solving skills
• Explain how a computer system works.	D. literacy and numeracy skills
	E. responsible behaviour to self, others and society
	F. interpersonal and communications skills

Course Learning Outcomes
(see Appendix A for more on learning outcomes)

At the end of the course, the successful student will know and be able to:
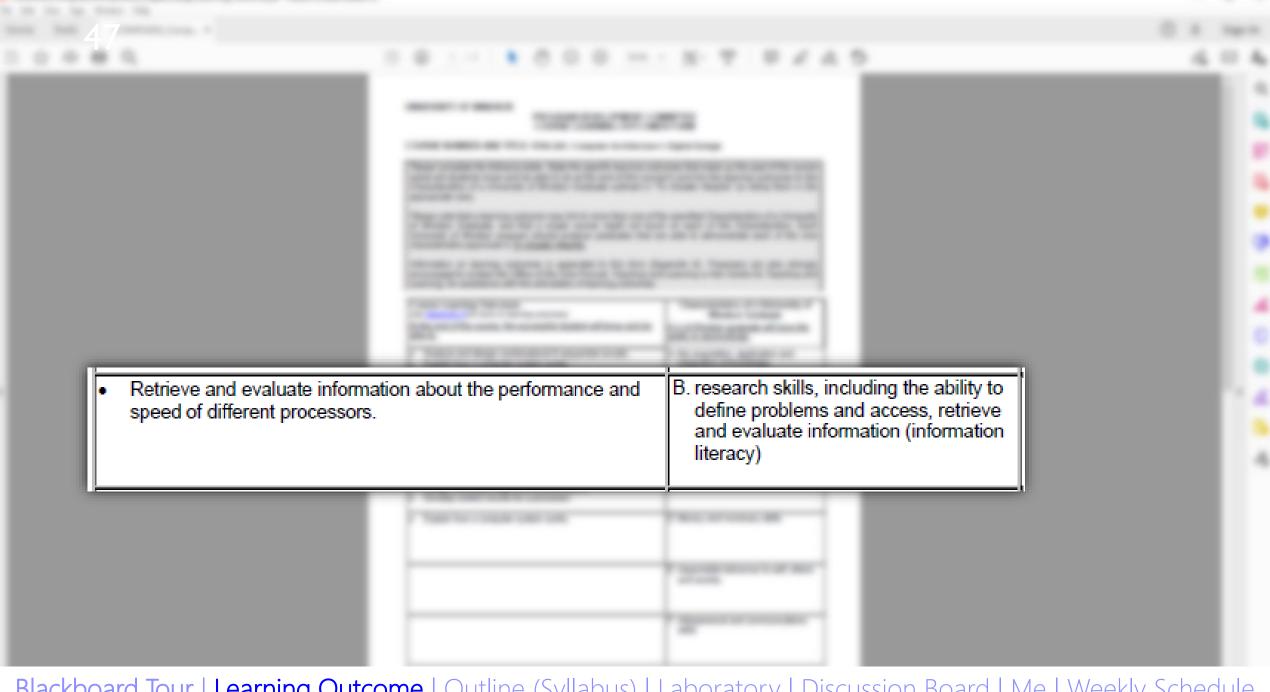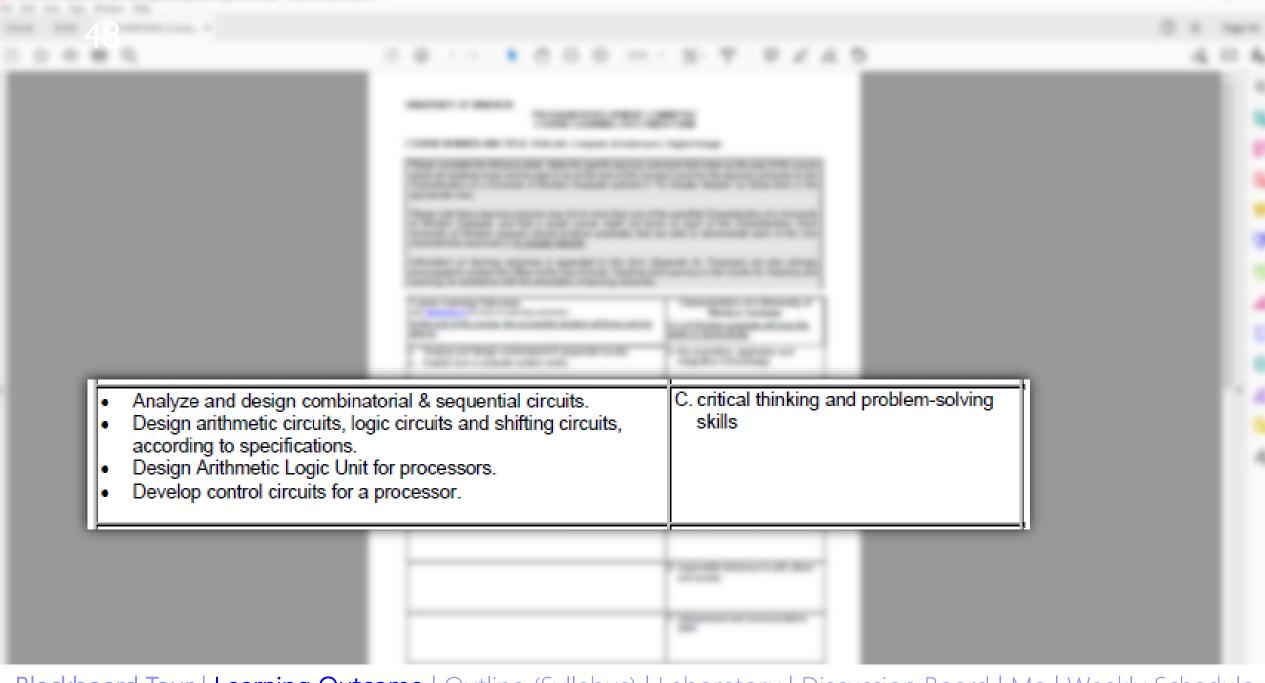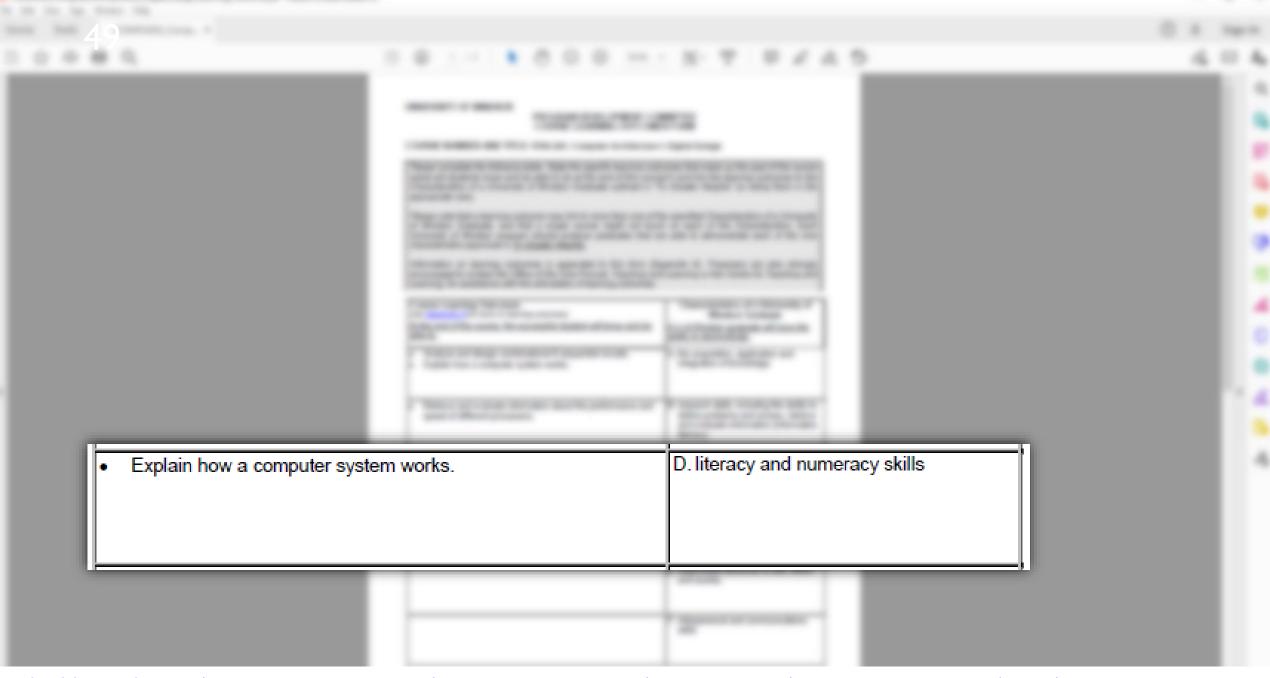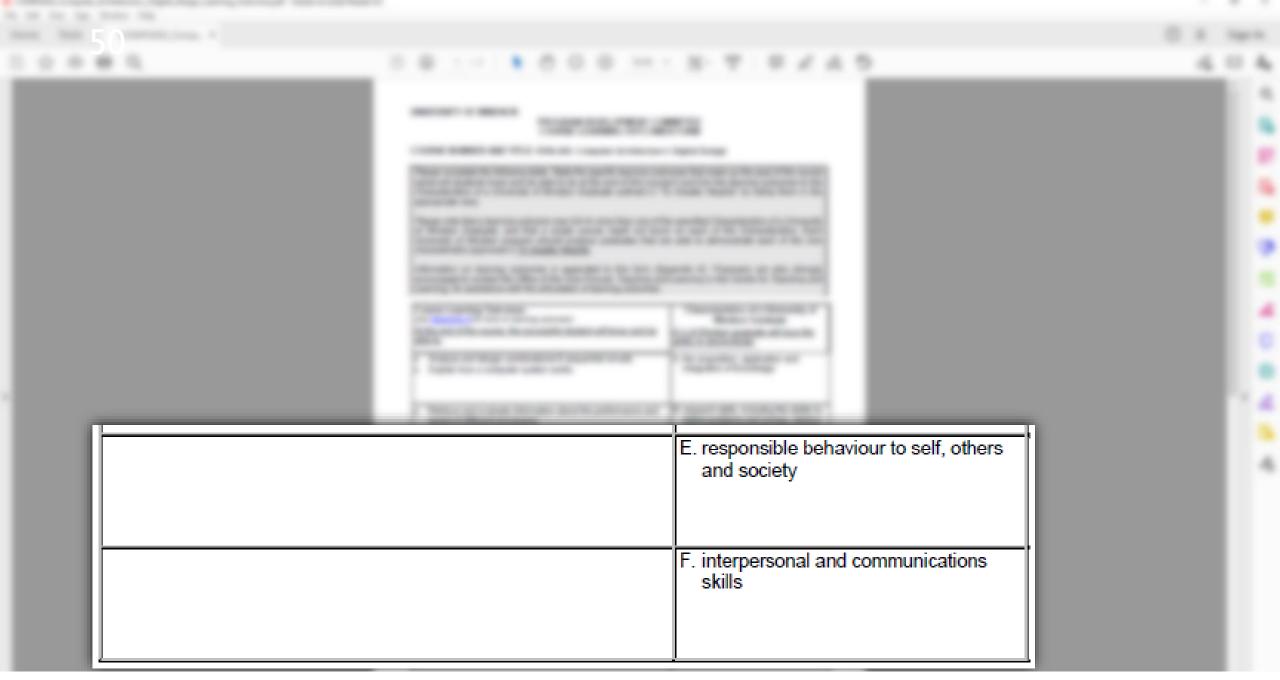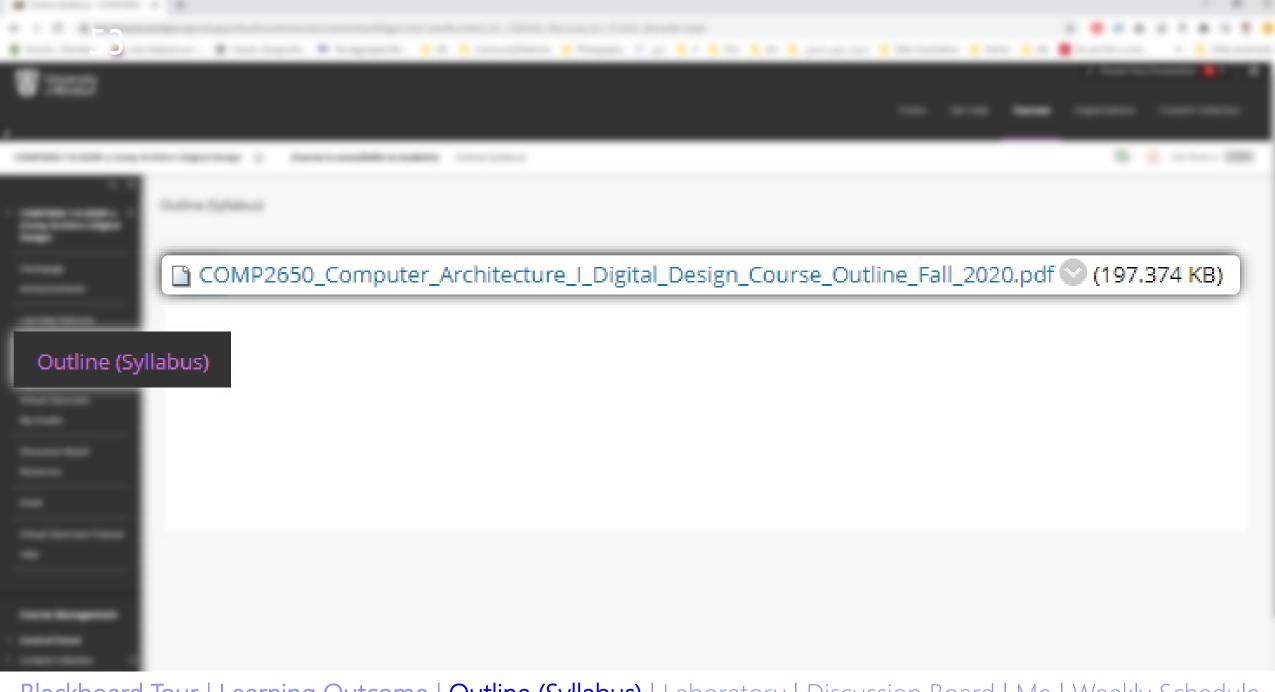
Blackboard Tour | Learning Outcome | Outline (Syllabus) | Laboratory | Discussion Board | Me | Weekly Schedule

Characteristics of a University of Windsor Graduate

A U of Windsor graduate will have the ability to demonstrate:

• Analyze and design combinatorial & sequential circuits. • Explain how a computer system works.	A. the acquisition, application and integration of knowledge

• Retrieve and evaluate information about the performance and speed of different processors.	B. research skills, including the ability to define problems and access, retrieve and evaluate information (information literacy)

• Analyze and design combinatorial & sequential circuits. • Design arithmetic circuits, logic circuits and shifting circuits, according to specifications. • Design Arithmetic Logic Unit for processors. • Develop control circuits for a processor.	C. critical thinking and problem-solving skills

Explain how a computer system works.	D. literacy and numeracy skills

	E. responsible behaviour to self, others and society
	F. interpersonal and communications skills

	G. teamwork, and personal and group leadership skills
	H. creativity and aesthetic appreciation
	I. the ability and desire for continuous learning

COURSE OUTLINE

aka. Course Syllabus

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
   |---> Outline (Syllabus)
     |---> COMP2650_Computer_Architecture_I_Digital_Design_Course_Outline.pdf

📄 COMP2650_Computer_Architecture_I_Digital_Design_Course_Outline_Fall_2020.pdf ⌄ (197.374 KB)

Outline (Syllabus)

University of Windsor
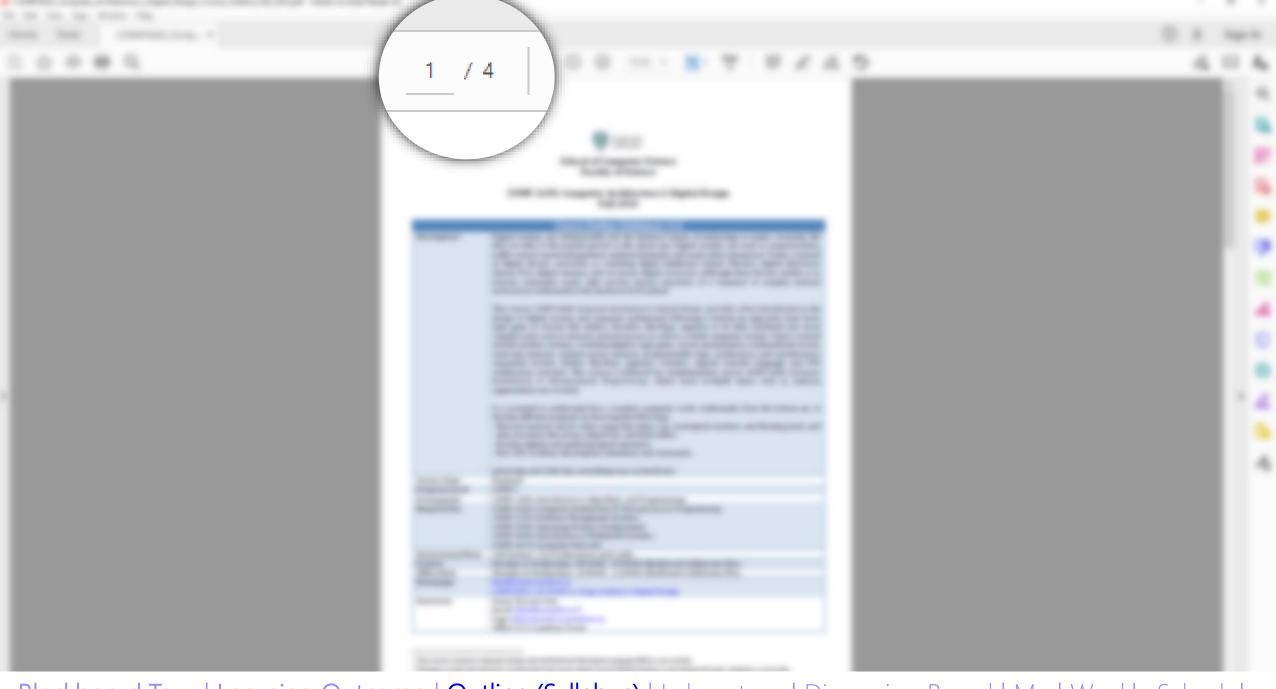
School of Computer Science
Faculty of Science
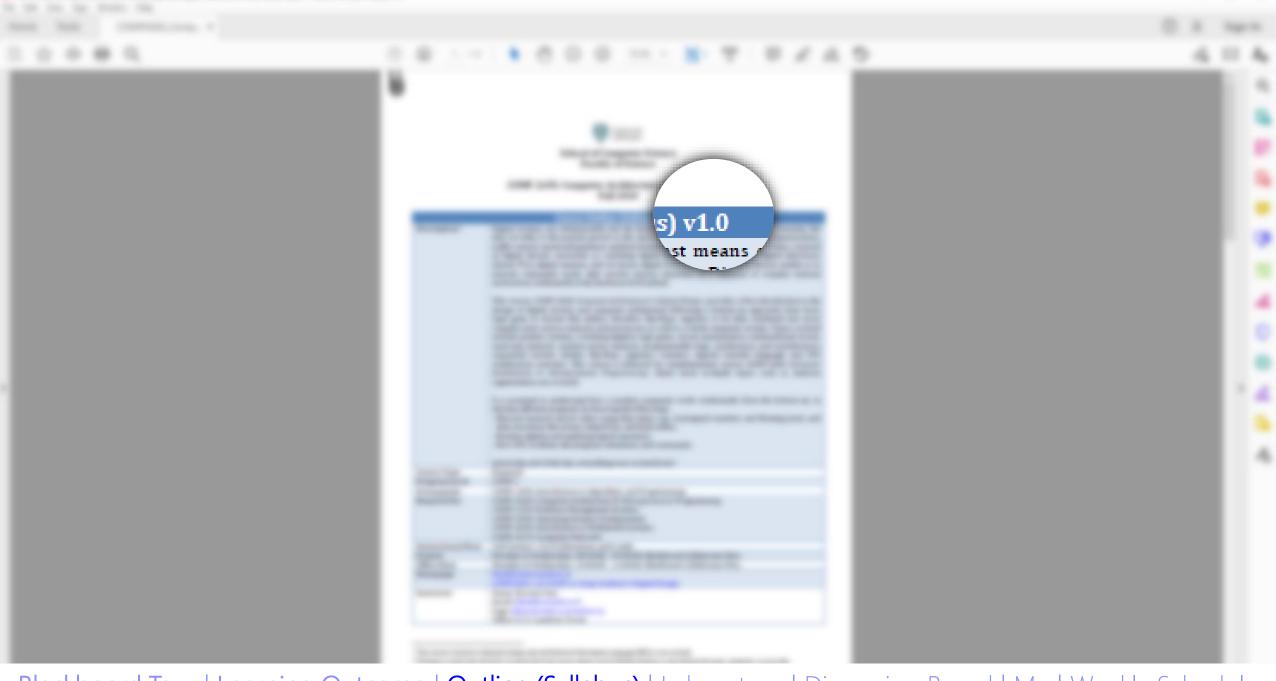
COMP-2650: Computer Architecture I: Digital Design
Fall 2020

Course Outline (Syllabus) v1.0
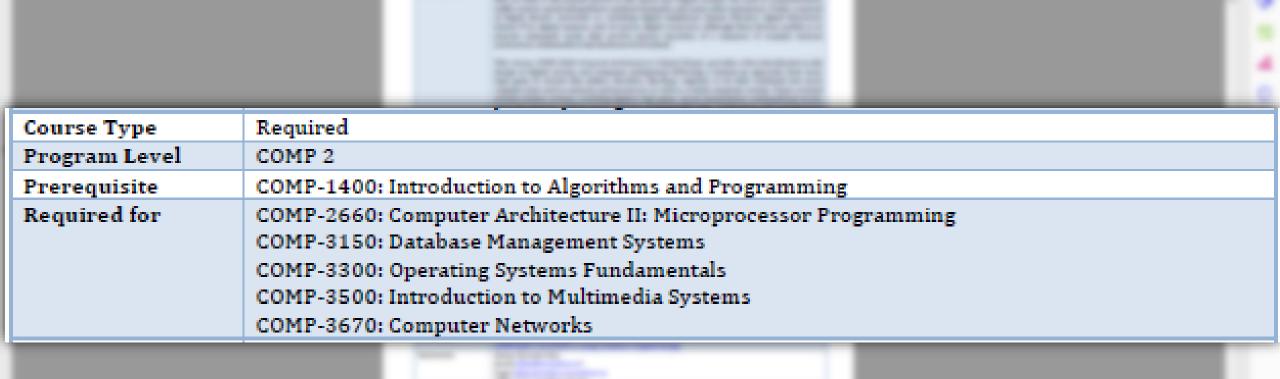
systems are indispensable and the foremost means of technol
refer to the present period as the *digital age*. Digital system
ntrol, spacecraft guidance, medical treatment, and many oth
devices surrounds us, including digital telephones (S
digital cameras, and, of course, digital *computers*.
nds easily, they involve precise executio
ath at the hardware level indeed.
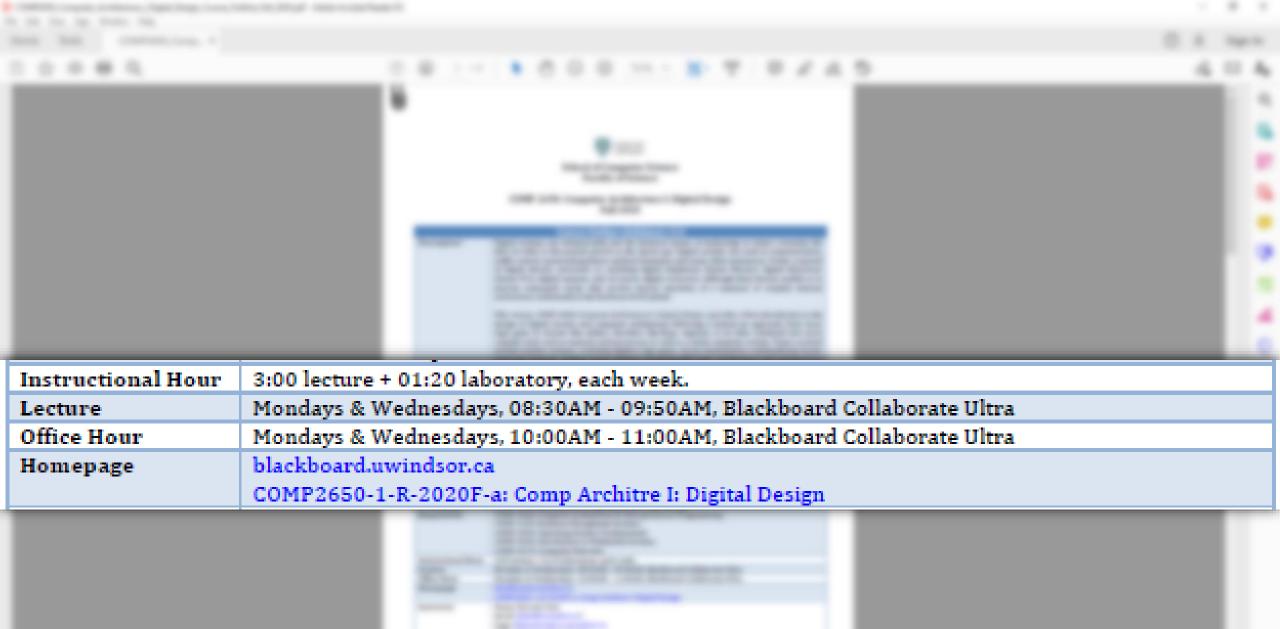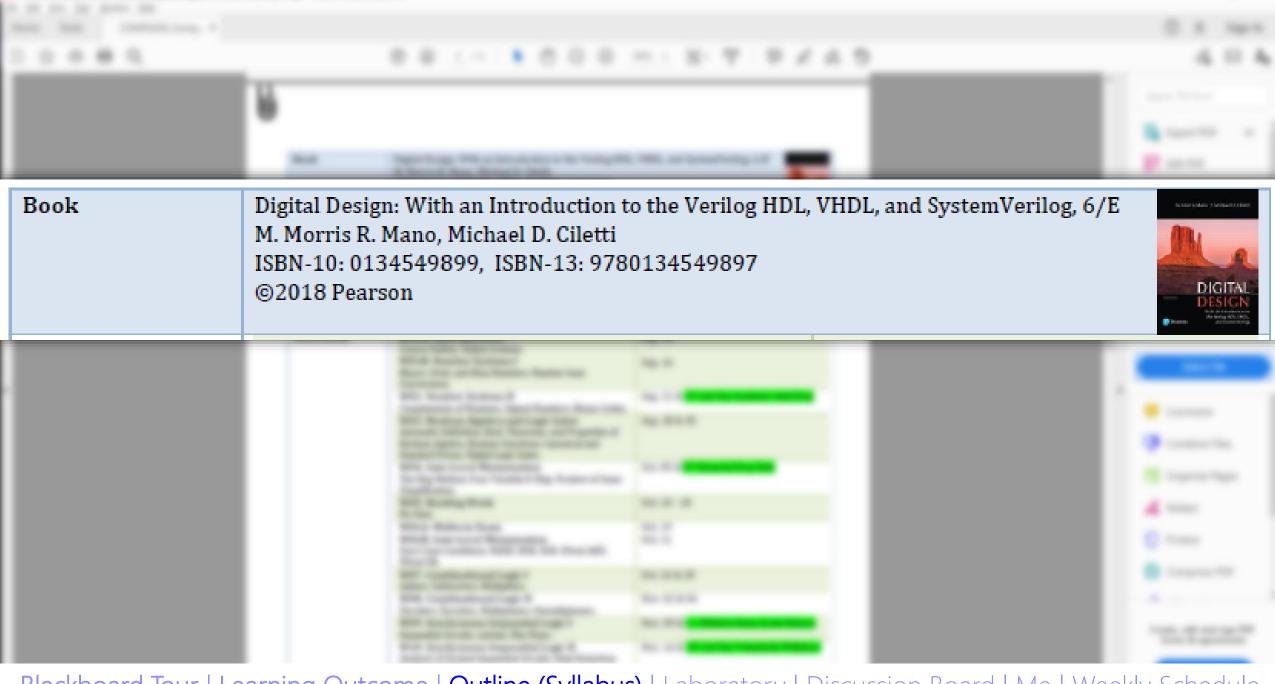
Course Type	Required
Program Level	COMP 2
Prerequisite	COMP-1400: Introduction to Algorithms and Programming
Required for	COMP-2660: Computer Architecture II: Microprocessor Programming
	COMP-3150: Database Management Systems
	COMP-3300: Operating Systems Fundamentals
	COMP-3500: Introduction to Multimedia Systems
	COMP-3670: Computer Networks

Instructional Hour	3:00 lecture + 01:20 laboratory, each week.
Lecture	Mondays & Wednesdays, 08:30AM - 09:50AM, Blackboard Collaborate Ultra
Office Hour	Mondays & Wednesdays, 10:00AM - 11:00AM, Blackboard Collaborate Ultra
Homepage	blackboard.uwindsor.ca COMP2650-1-R-2020F-a: Comp Architre I: Digital Design

Book	Digital Design: With an Introduction to the Verilog HDL, VHDL, and SystemVerilog, 6/E
	M. Morris R. Mano, Michael D. Ciletti
	ISBN-10: 0134549899, ISBN-13: 9780134549897
	©2018 Pearson
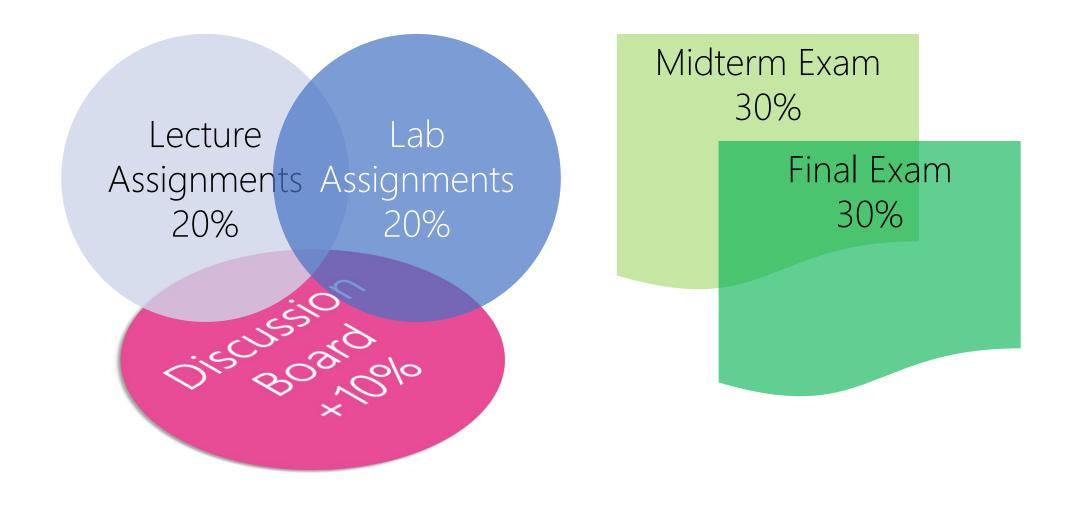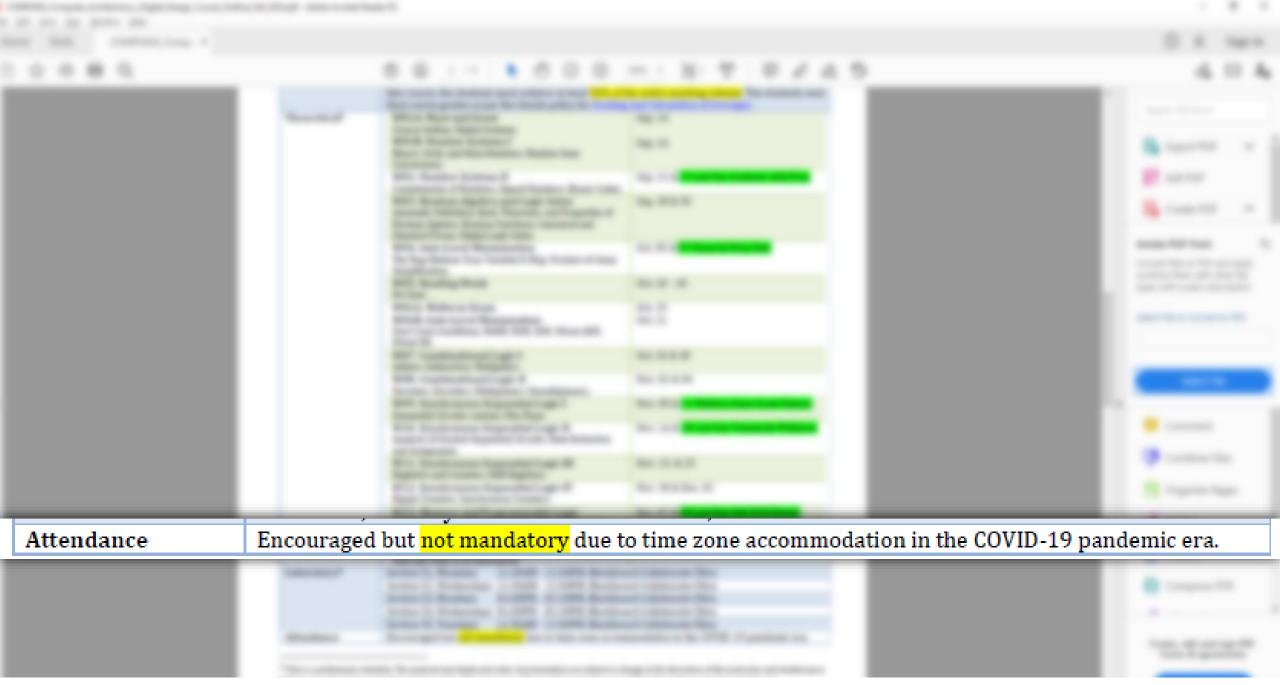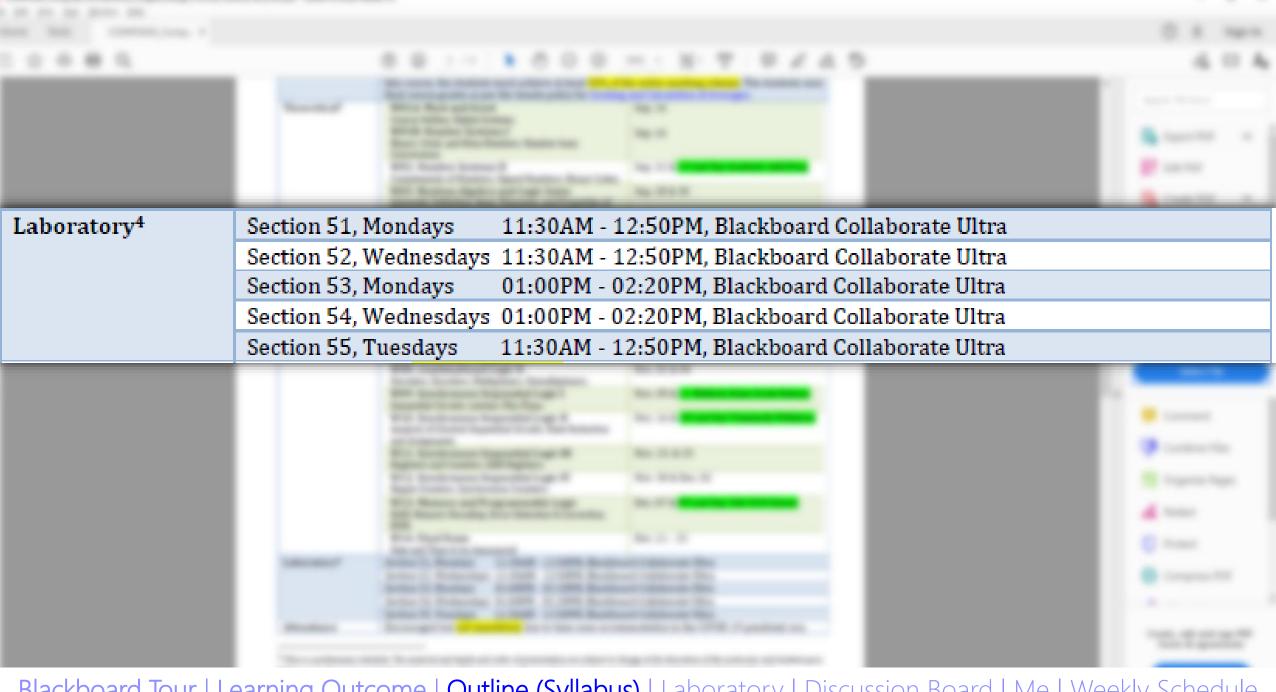
Marking Scheme	Lecture (Weekly) Assignments	20%
	Lab Assignments	20%
	Midterm Exam	30%
	Final Exam	30%
	Discussion Board (Bonus+)	10%
Remarks	The written reports will be assessed not only on their academic merit but also on the communication skills of the author as exhibited through the reports. To achieve a passing grade in this course, the students must achieve at least 50% of the entire marking scheme. The students earn final course grades as per the Senate policy for Grading and Calculation of Averages.	

Lecture Assignments 20%

Lab Assignments 20%

Discussion Board +10%

Midterm Exam 30%

Final Exam 30%

Theoretical[3]		
W01A: Meet and Greet Course Outline. Digital Systems.	Sep. 14	
W01B: Number Systems I Binary, Octal, and Hexa Numbers. Number-base Conversions.	Sep. 16	
W02: Number Systems II Complements of Numbers. Signed Numbers. Binary Codes.	Sep. 21 & 23 Last Day Academic Add/Drop	
W03: Boolean Algebra and Logic Gates Axiomatic Definition, Basic Theorems, and Properties of Boolean Algebra. Boolean Functions. Canonical and Standard Forms. Digital Logic Gates.	Sep. 28 & 30	
W04: Gate-Level Minimization The Map Method. Four-Variable K-Map. Product-of-Sums Simplification.	Oct. 05 & 07 Financial Drop Date	
W05: Reading Week No Class	Oct. 10 - 18	
W06A: Midterm Exam **W06B: Gate-Level Minimization** Don't-Care Conditions. NAND, NOR, XOR, Wired AND, Wired OR.	Oct. 19 Oct. 21	
W07: Combinational Logic I Adders. Subtractors. Multipliers.	Oct. 26 & 28	
W08: Combinational Logic II Decoders. Encoders. Multiplexers. Demultiplexers.	Nov. 02 & 04	
W09: Synchronous Sequential Logic I Sequential Circuits. Latches. Flip-Flops.	Nov. 09 & 11 Midterm Exam Grade Release	
W10: Synchronous Sequential Logic II Analysis of Clocked Sequential Circuits. State Reduction and Assignment.	Nov. 16 & 18 Last Day Voluntarily Withdraw	
W11: Synchronous Sequential Logic III Registers and Counters. Shift Registers.	Nov. 23. & 25	
W12: Synchronous Sequential Logic IV Ripple Counters. Synchronous Counters.	Nov. 30 & Dec. 02	
W13: Memory and Programmable Logic RAM, Memory Decoding. Error Detection & Correction. ROM.	Dec. 07 & 09 Last Day: Fall 2020 Classes	
W14: Final Exam	Dec. 11 – 22	
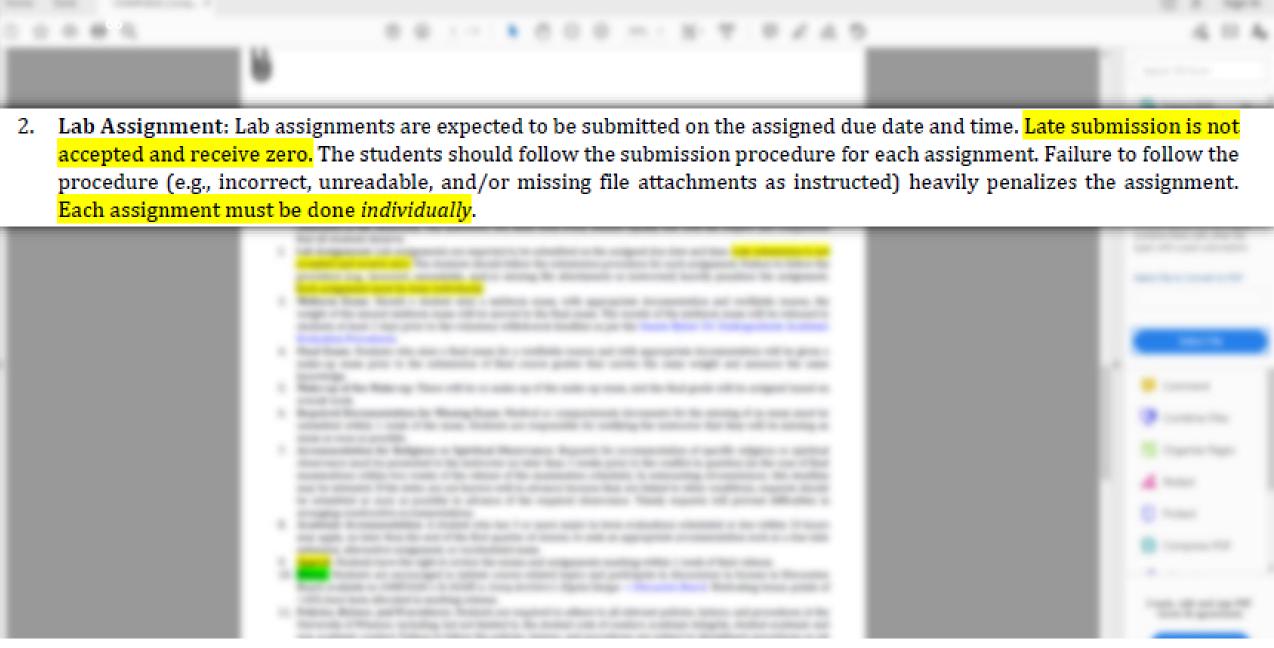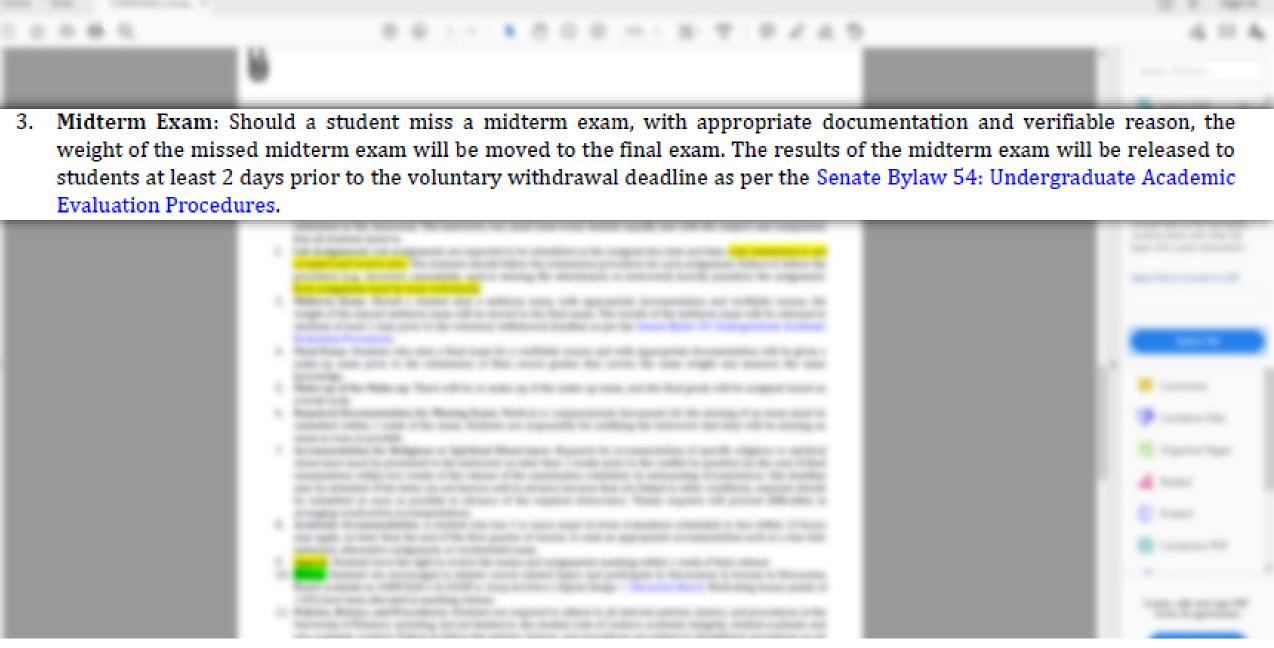
| Attendance | Encouraged but not mandatory due to time zone accommodation in the COVID-19 pandemic era. |

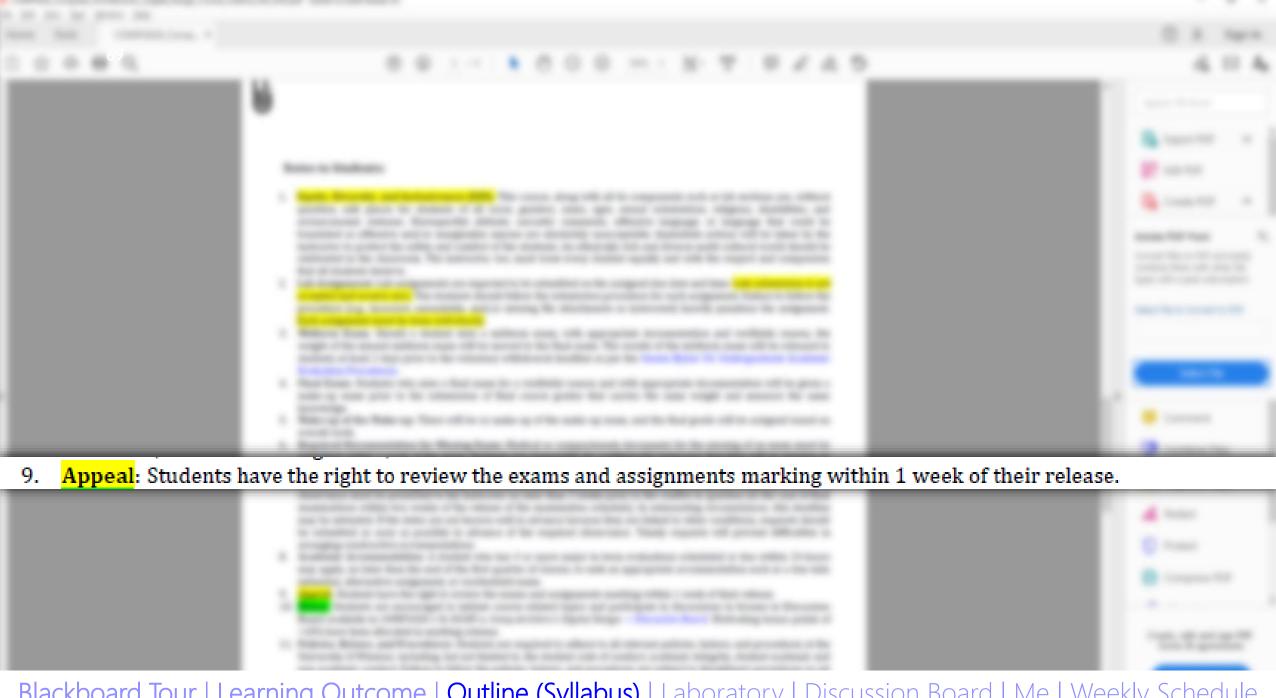| Laboratory[4]	Section 51, Mondays	11:30AM - 12:50PM, Blackboard Collaborate Ultra
	Section 52, Wednesdays	11:30AM - 12:50PM, Blackboard Collaborate Ultra
	Section 53, Mondays	01:00PM - 02:20PM, Blackboard Collaborate Ultra
	Section 54, Wednesdays	01:00PM - 02:20PM, Blackboard Collaborate Ultra
	Section 55, Tuesdays	11:30AM - 12:50PM, Blackboard Collaborate Ultra

Notes to Students:

1. **Equity, Diversity, and Inclusiveness (EDI):** This course, along with all its components such as lab sections are, without question, safe places for students of all races, genders, sexes, ages, sexual orientations, religions, disabilities, and socioeconomic statuses. Disrespectful attitude, sarcastic comments, offensive language, or language that could be translated as offensive and/or marginalize anyone are absolutely unacceptable. Immediate actions will be taken by the instructor to protect the safety and comfort of the students. An ethnically rich and diverse multi-cultural world should be celebrated in the classroom. The instructor, too, must treat every student equally and with the respect and compassion that all students deserve.
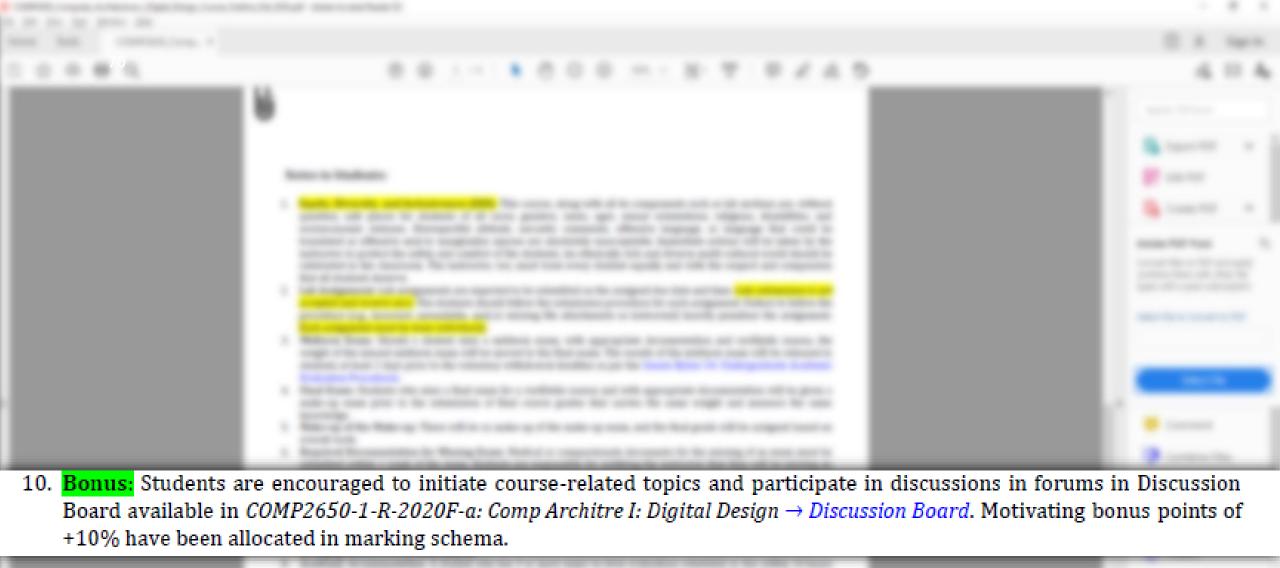
2. **Lab Assignment:** Lab assignments are expected to be submitted on the assigned due date and time. ==Late submission is not accepted and receive zero.== The students should follow the submission procedure for each assignment. Failure to follow the procedure (e.g., incorrect, unreadable, and/or missing file attachments as instructed) heavily penalizes the assignment. ==Each assignment must be done *individually*==.

3. **Midterm Exam:** Should a student miss a midterm exam, with appropriate documentation and verifiable reason, the weight of the missed midterm exam will be moved to the final exam. The results of the midterm exam will be released to students at least 2 days prior to the voluntary withdrawal deadline as per the Senate Bylaw 54: Undergraduate Academic Evaluation Procedures.

4. **Final Exam**: Students who miss a final exam for a verifiable reason and with appropriate documentation will be given a make-up exam prior to the submission of final course grades that carries the same weight and measure the same knowledge.

5. **Make-up of the Make-up**: There will be *no* make-up of the make-up exam, and the final grade will be assigned based on overall work.

6. **Required Documentation for Missing Exam**: Medical or compassionate documents for the missing of an exam must be submitted within 1 week of the exam. Students are responsible for notifying the instructor that they will be missing an exam as soon as possible.

7. **Accommodation for Religious or Spiritual Observance**: Requests for accommodation of specific religious or spiritual observance must be presented to the instructor no later than 2 weeks prior to the conflict in question (in the case of final examinations within two weeks of the release of the examination schedule). In extenuating circumstances, this deadline may be extended. If the dates are not known well in advance because they are linked to other conditions, requests should be submitted as soon as possible in advance of the required observance. Timely requests will prevent difficulties in arranging constructive accommodations.

8. **Academic Accommodation**: A student who has 3 or more major in-term evaluations scheduled or due within 24 hours may apply, no later than the end of the first quarter of classes, to seek an appropriate accommodation such as a due date extension, alternative assignment, or rescheduled exam.

Notes to Students:

1. **Equity, Diversity, and Inclusiveness (EDI):** This course, along with all its components such as lab sections are, without exception, safe places for students of all races, genders, sexes, ages, sexual orientations, religions, disabilities, and environments. Without disrespectful attitude towards commonly offensive language or language that could be translated to offensive and/or inappropriate expresses are absolutely unacceptable, responsible actions will be taken by the instructor to protect the safety and comfort of the students. An ethnically rich and diverse multi-cultural world should be celebrated in the classroom. The instructor, too, must treat every student equally and with the respect and compassion that all students deserve.

2. Late assignment, Late components are expected to be submitted on the assigned due date and time. **Late submissions are accepted and result are.** The instructor should follow the submission procedure for each assignment. Failure to follow the procedure may decrease accountability, and/or solving the attachments or unreviewed records penalizes the assignment. **Late assignments must be done within 24 hours.**

3. Midterm Exam: There is a limited time / uniform exam, with appropriate documentation and verifiable reason, the weight of the missed midterm exam will be moved to the final mark. The result of the midterm exam will be released to students at least 7 days prior to the voluntary withdrawal deadline as per the **Senate Rules for Undergraduate Academic Evaluation Procedures.**

4. Final Exam: Students who miss a final exam for a verifiable reason and with appropriate documentation will be given a make-up exam prior to the submission of final course grades that carries the same weight and measure the same knowledge.

5. Make-up of the Make-up: There will be no make-up of the make-up exam, and the final grade will be computed based on course load.

6. Required Documentation for Missing Exam: Medical or compassionate documents for the missing of an exam must be...

9. **Appeal**: Students have the right to review the exams and assignments marking within 1 week of their release.

10. **Bonus:** Students are encouraged to initiate course-related topics and participate in discussions in forums in Discussion Board available in *COMP2650-1-R-2020F-a: Comp Architre I: Digital Design → Discussion Board*. Motivating bonus points of +10% have been allocated in marking schema.

11. **Policies, Bylaws, and Procedures:** Students are required to adhere to all relevant policies, bylaws, and procedures at the University of Windsor, including, but not limited to, the student code of conduct, academic integrity, student academic and non-academic conduct. Failure to follow the policies, bylaws, and procedures are subject to disciplinary procedures as set out under, but not limited to, the Senate Bylaw 31: Academic Integrity and Procedures for Addressing Student Non-Academic Misconduct. Regarding the plagiarism, the Blackboard's SafeAssign will be used for some or all student assignments or equivalent at the instructor's discretion. Plagiarized submissions or equivalent (e.g., exams), i.e., submissions with the same or minor modifications, receive zero. Should you need to record the lectures, please follow the Senate Policy on Recording Lectures.

12. **Communication:** Students are required to obtain and maintain a University of Windsor e-mail account, [uwindid]@uwindsor.ca, for timely communications with the instructor. The course homepage on the Blackboard, *COMP2650-1-R-2020F-a: Comp Architre I: Digital Design*, is the main notification center for the course announcements and repository for the course material and resources. Blackboard Collaborate Ultra at *COMP2650-1-R-2020F-a: Comp Architre I: Digital Design* → *Virtual Classroom* → *Course Room* is the official place for the lectures and office hours with the instructor. In Microsoft Teams, the team *SCS - COMP2650 - Fall2020* is provided for emergency cases and backup plans only.

13. **Change Notification:** Any changes in the course outline, exam dates, marking, or evaluation will be discussed in class at least 2 weeks prior to being implemented.

14. **Student Evaluation of Teaching (SET):** The Student Evaluation of Teaching (SET) will be conducted during the last 2 weeks of the classes.

15. **Online Experience:** Participants in online lectures and lab sections include an instructor, a moderator, and students. Students are able to share camera or send messages but cannot share audio unless they Raise Hand, and the moderator or the instructor allows them. The moderator also supervises private messages. Students are encouraged to let the moderator and/or the instructor know of any connection issues asap regarding the quality of presentation in terms of audio and video (e.g., slides).

16. **Feeling Overwhelmed?** Should face obstacles and experience difficulties that affect her academic performance, students can reach out to the following service centers as well as other on- and off-campus resources listed at www.uwindsor.ca/wellness:

- Student Health Services (www.uwindsor.ca/studenthealthservices/)
- Student Counselling Centre (www.uwindsor.ca/studentcounselling/)
- Peer Support Centre
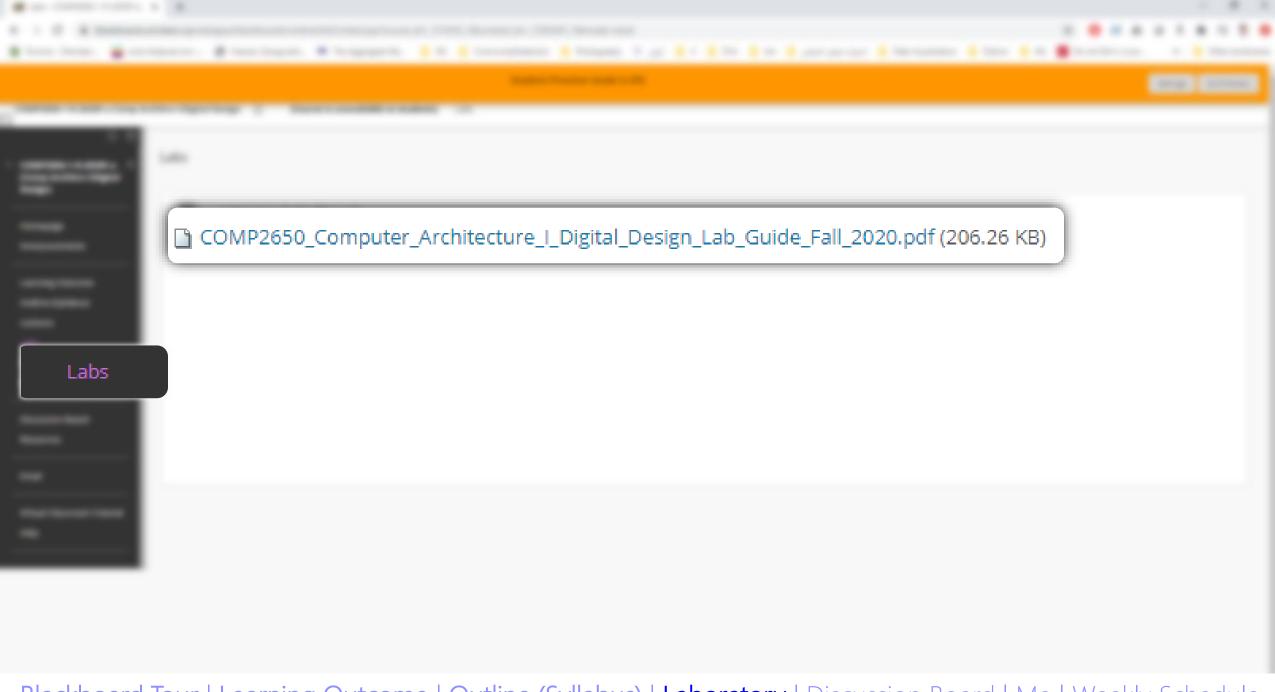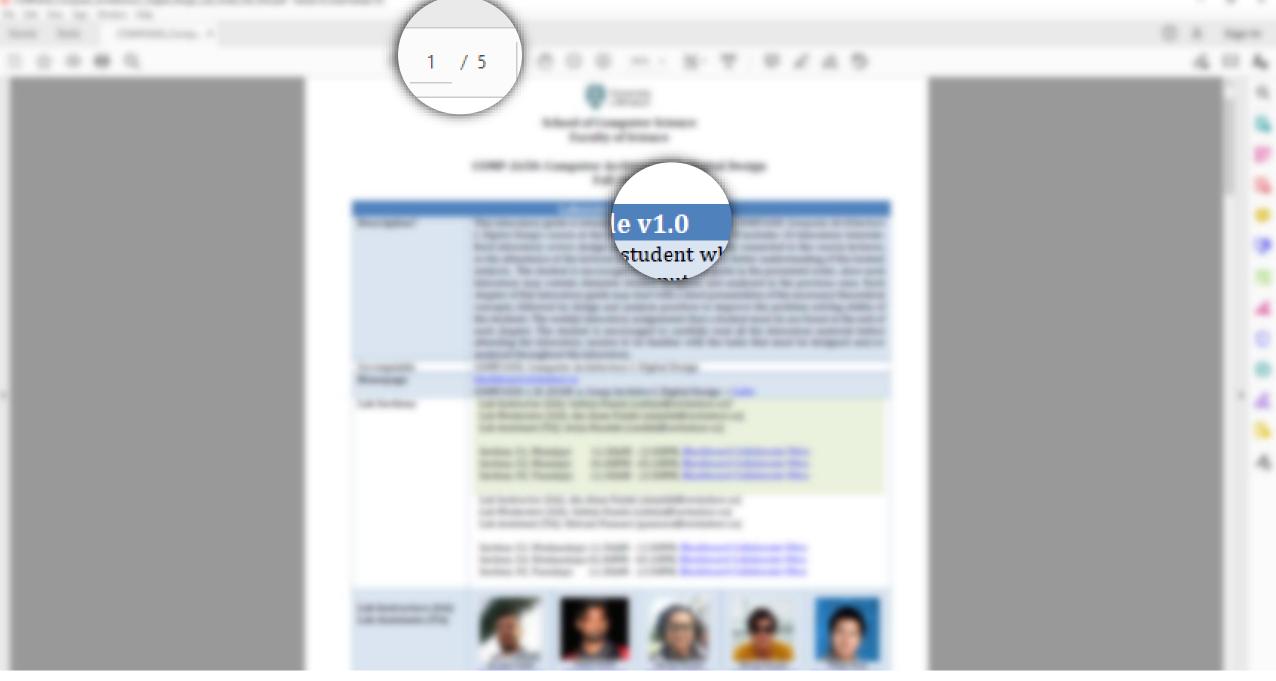
LABORATORY GUIDE

aka. Lab Manual

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
    |---> Labs
        |---> COMP2650_Computer_Architecture_I_Digital_Design_Lab_Guide.pdf
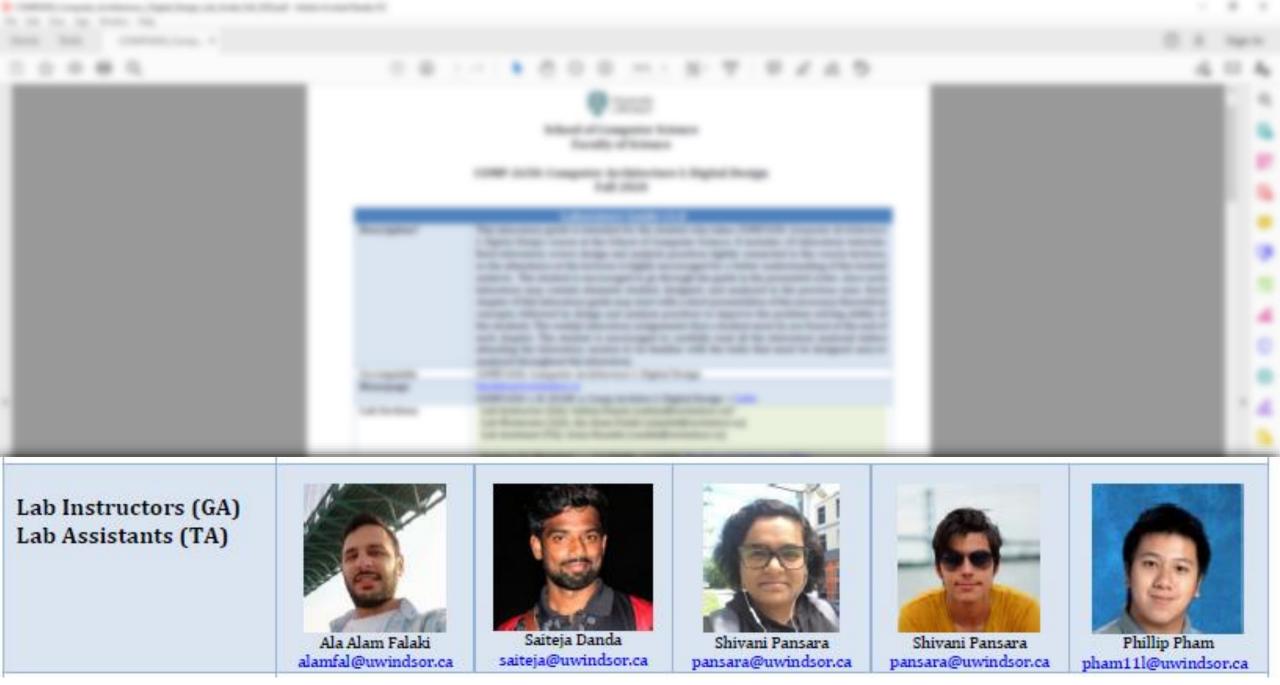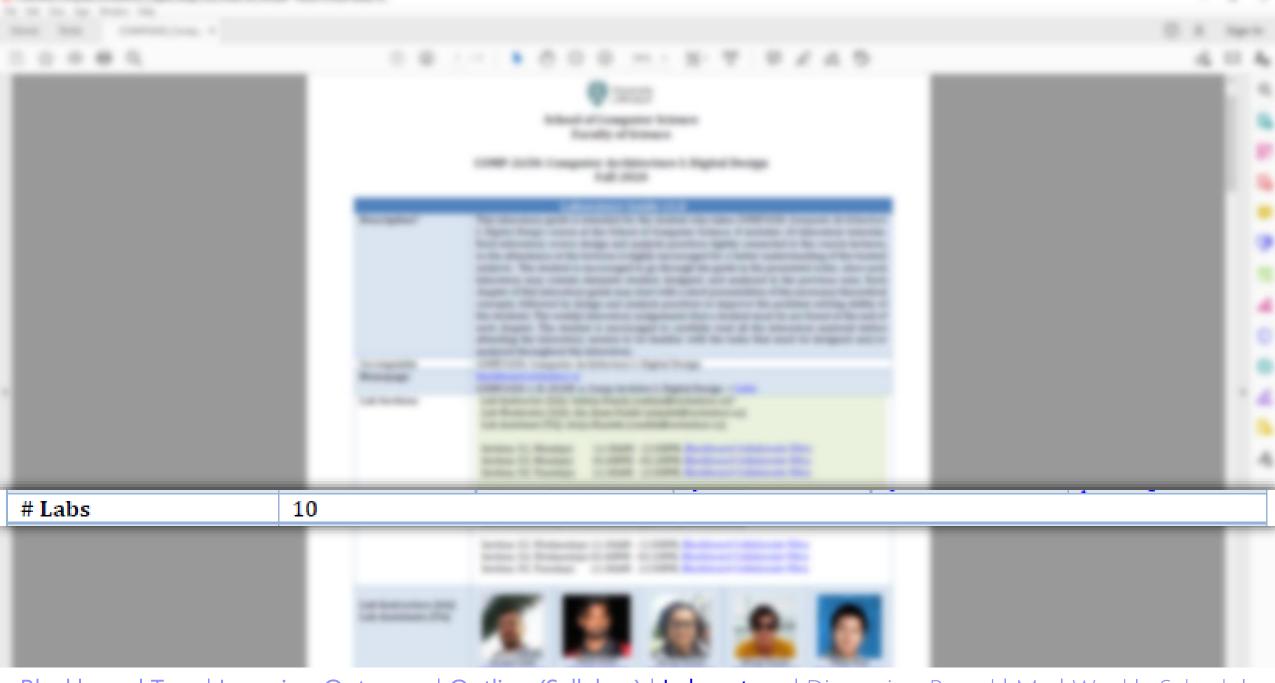
Labs

📄 COMP2650_Computer_Architecture_I_Digital_Design_Lab_Guide_Fall_2020.pdf (206.26 KB)

Labs

Lab Instructors (GA)
Lab Assistants (TA)

Ala Alam Falaki	**Saiteja Danda**	**Shivani Pansara**	**Shivani Pansara**	**Phillip Pham**
alamfal@uwindsor.ca	saiteja@uwindsor.ca	pansara@uwindsor.ca	pansara@uwindsor.ca	pham11l@uwindsor.ca
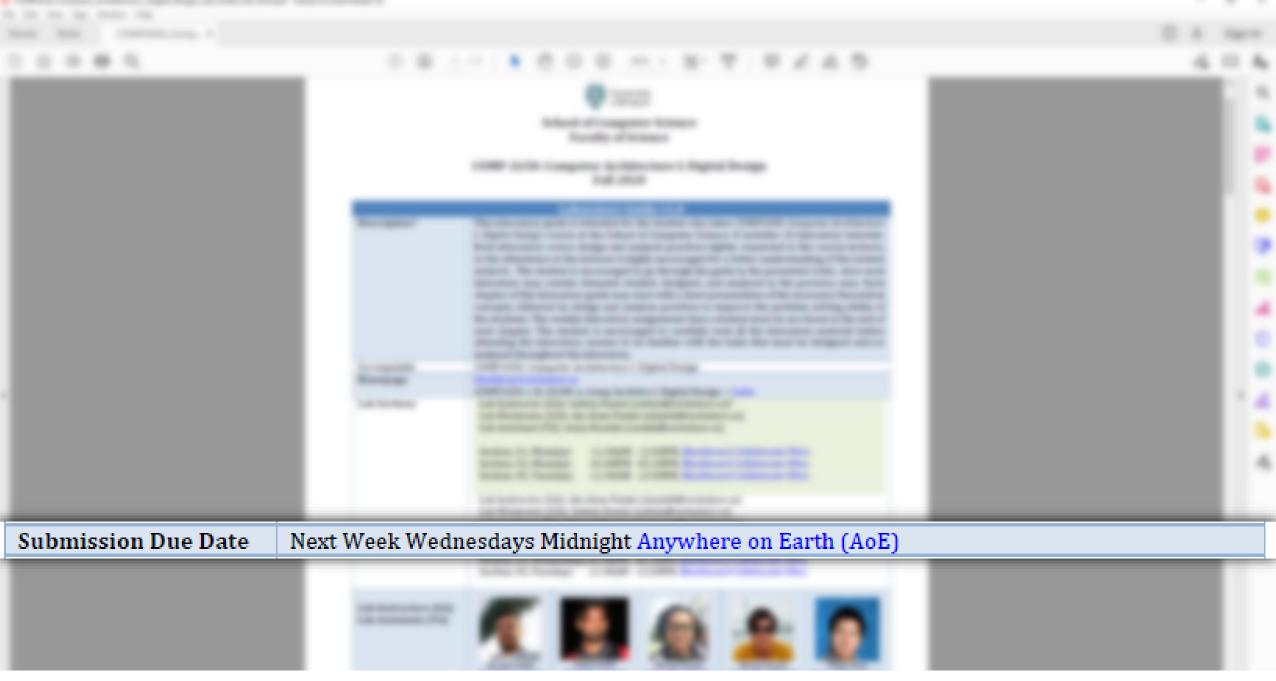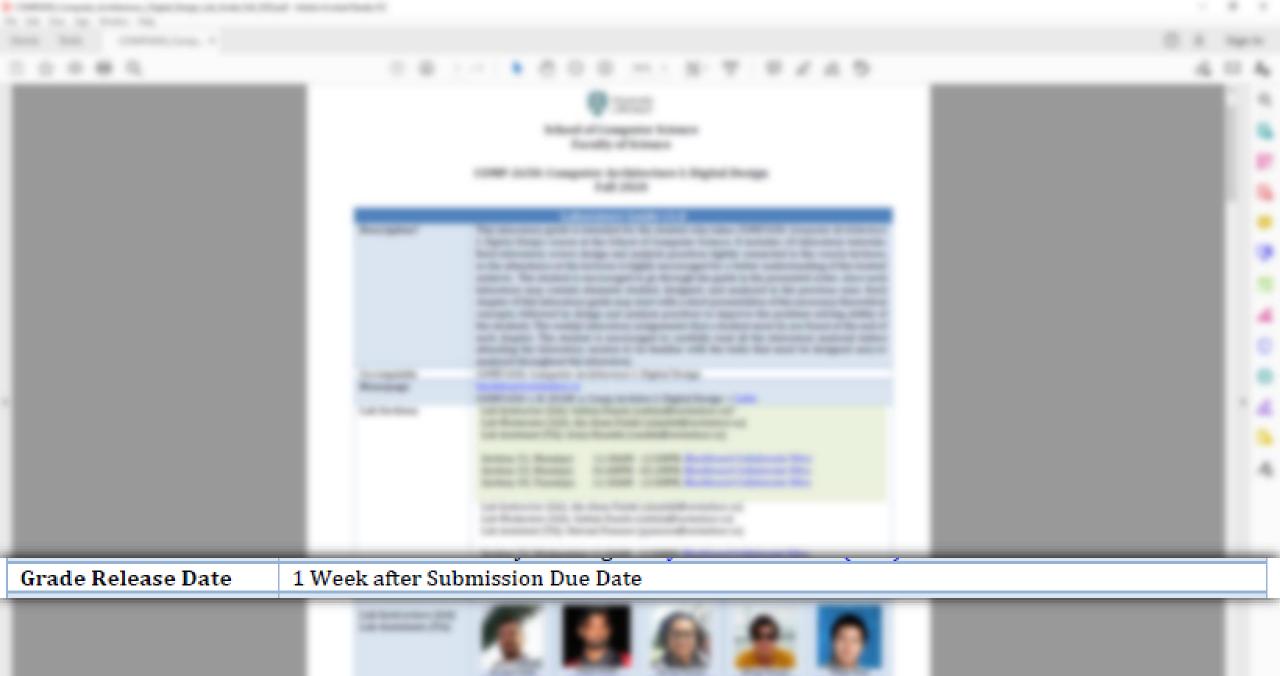
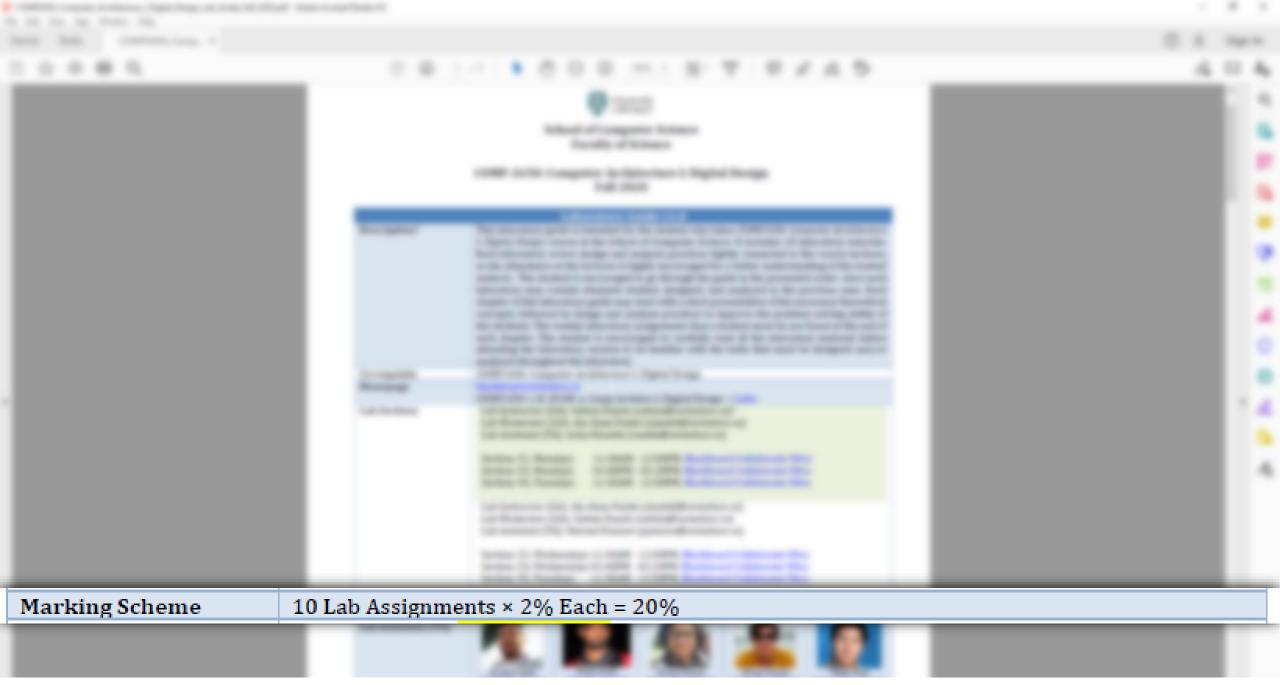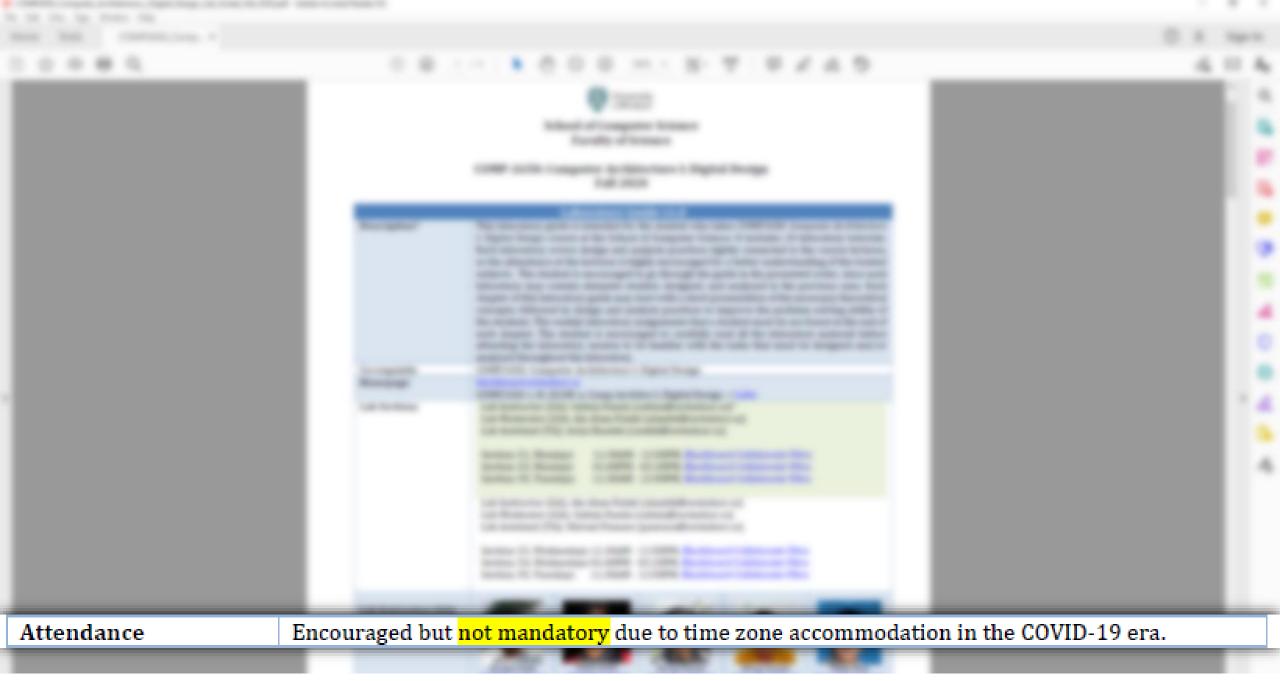| Lab Sections	Lab Instructor (GA): Saiteja Danda (saiteja@uwindsor.ca)[2]
	Lab Moderator (GA): Ala Alam Falaki (alamfal@uwindsor.ca)
	Lab Assistant (TA): Ariya Rasekh (rasekh@uwindsor.ca)
	Section: 51, Mondays 11:30AM - 12:50PM, Blackboard Collaborate Ultra
	Section: 53, Mondays 01:00PM - 02:20PM, Blackboard Collaborate Ultra
	Section: 55, Tuesdays 11:30AM - 12:50PM, Blackboard Collaborate Ultra
	Lab Instructor (GA): Ala Alam Falaki (alamfal@uwindsor.ca)
	Lab Moderator (GA): Saiteja Danda (saiteja@uwindsor.ca)
	Lab Assistant (TA): Shivani Pansara (pansara@uwindsor.ca)
	Section: 52, Wednesdays 11:30AM - 12:50PM, Blackboard Collaborate Ultra
	Section: 54, Wednesdays 01:00PM - 02:20PM, Blackboard Collaborate Ultra
	Section: 55, Tuesdays 11:30AM - 12:50PM, Blackboard Collaborate Ultra

# Labs	10

Submission Due Date	Next Week Wednesdays Midnight Anywhere on Earth (AoE)

Grade Release Date	1 Week after Submission Due Date

Marking Scheme	10 Lab Assignments × 2% Each = 20%

| Attendance | Encouraged but ==not mandatory== due to time zone accommodation in the COVID-19 era. |

Office Hours	N/A.
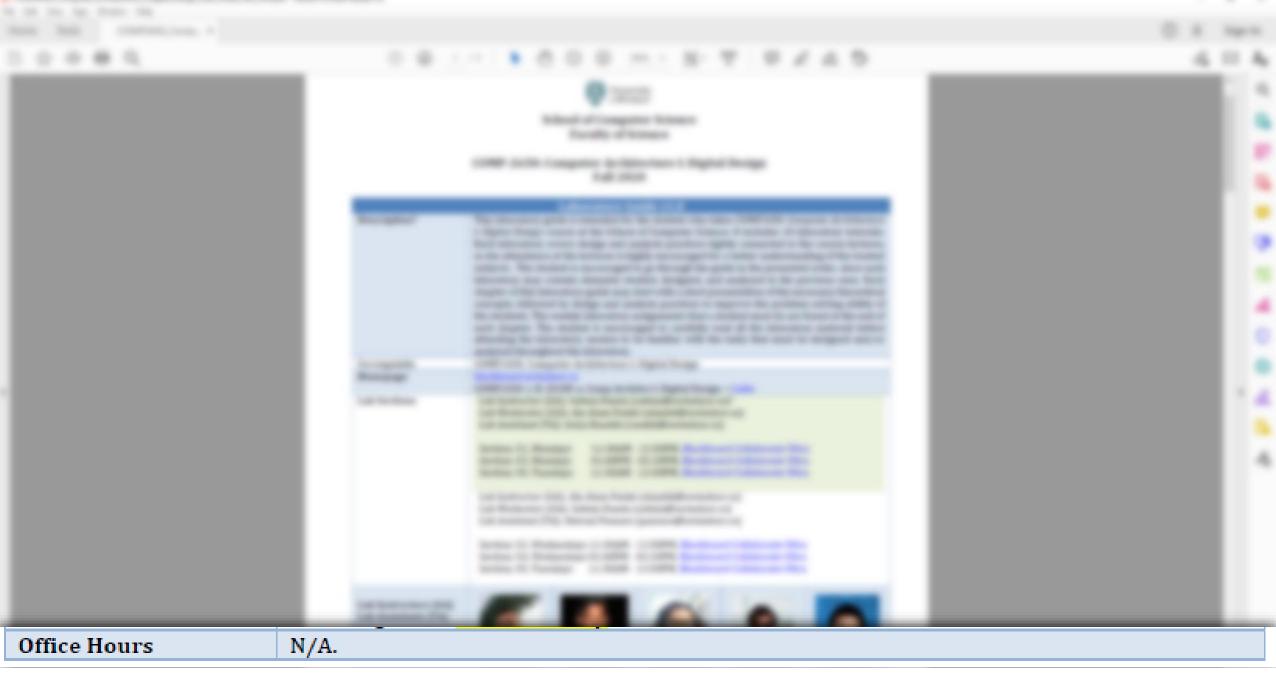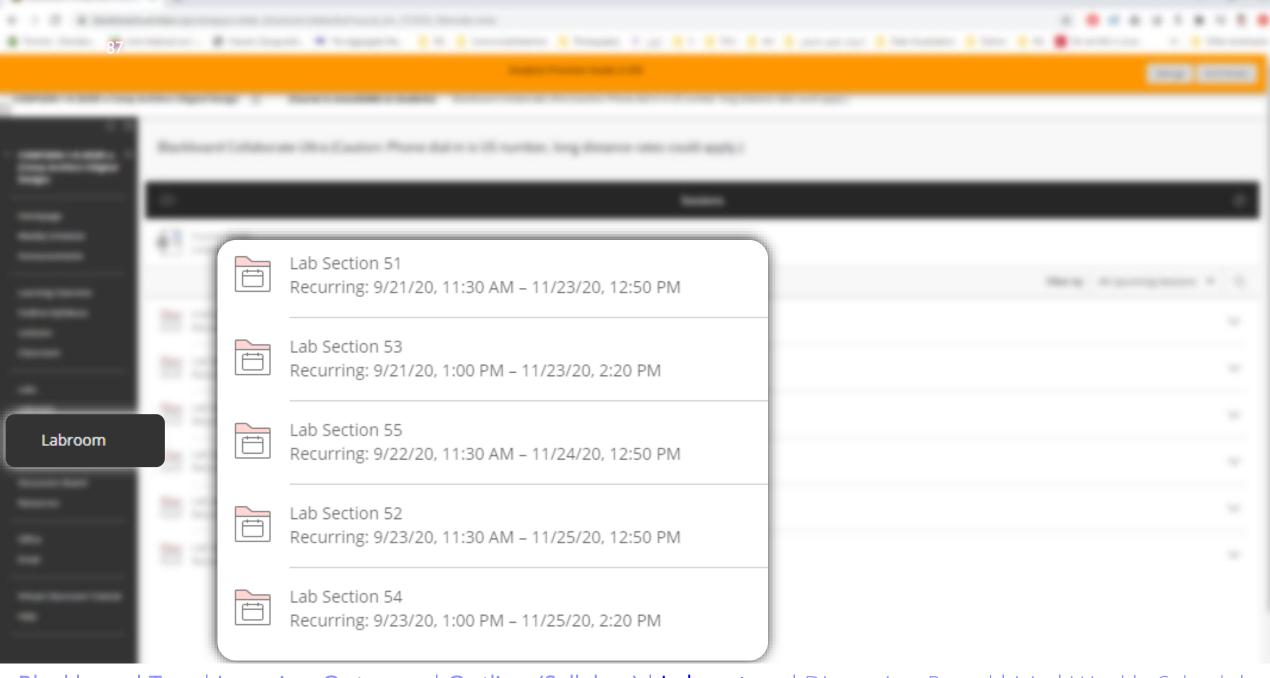
LABROOM

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
   |---> Labs
      |---> Labroom

Labroom

Lab Section 51
Recurring: 9/21/20, 11:30 AM – 11/23/20, 12:50 PM

Lab Section 53
Recurring: 9/21/20, 1:00 PM – 11/23/20, 2:20 PM

Lab Section 55
Recurring: 9/22/20, 11:30 AM – 11/24/20, 12:50 PM

Lab Section 52
Recurring: 9/23/20, 11:30 AM – 11/25/20, 12:50 PM

Lab Section 54
Recurring: 9/23/20, 1:00 PM – 11/25/20, 2:20 PM

Blackboard Tour | Learning Outcome | Outline (Syllabus) | **Laboratory** | Discussion Board | Me | Weekly Schedule
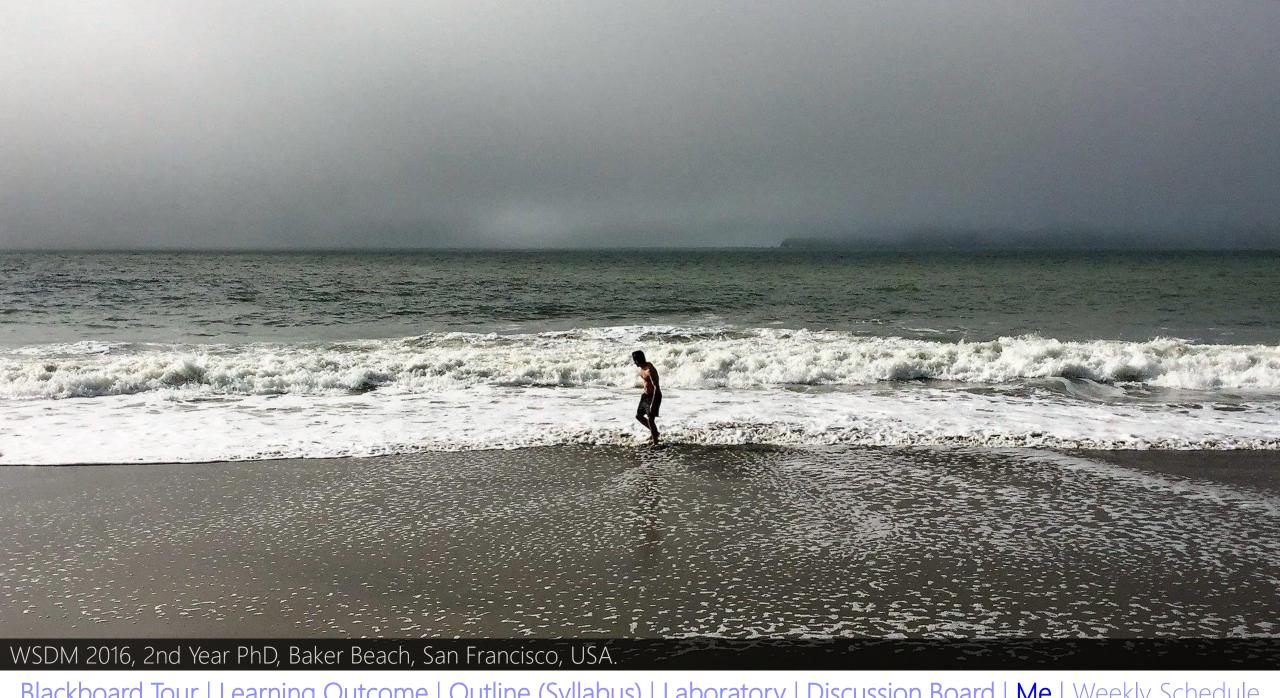
DISCUSSION BOARD

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
   |---> Discussion Board

FORUM	DESCRIPTION	TOTAL POSTS
Midterm Exam		0
Laboratory		0
Lectures		0
Final Exam		0
Miscellaneous		0

Discussion Board

ME

hfani.myweb.cs.uwindsor.ca
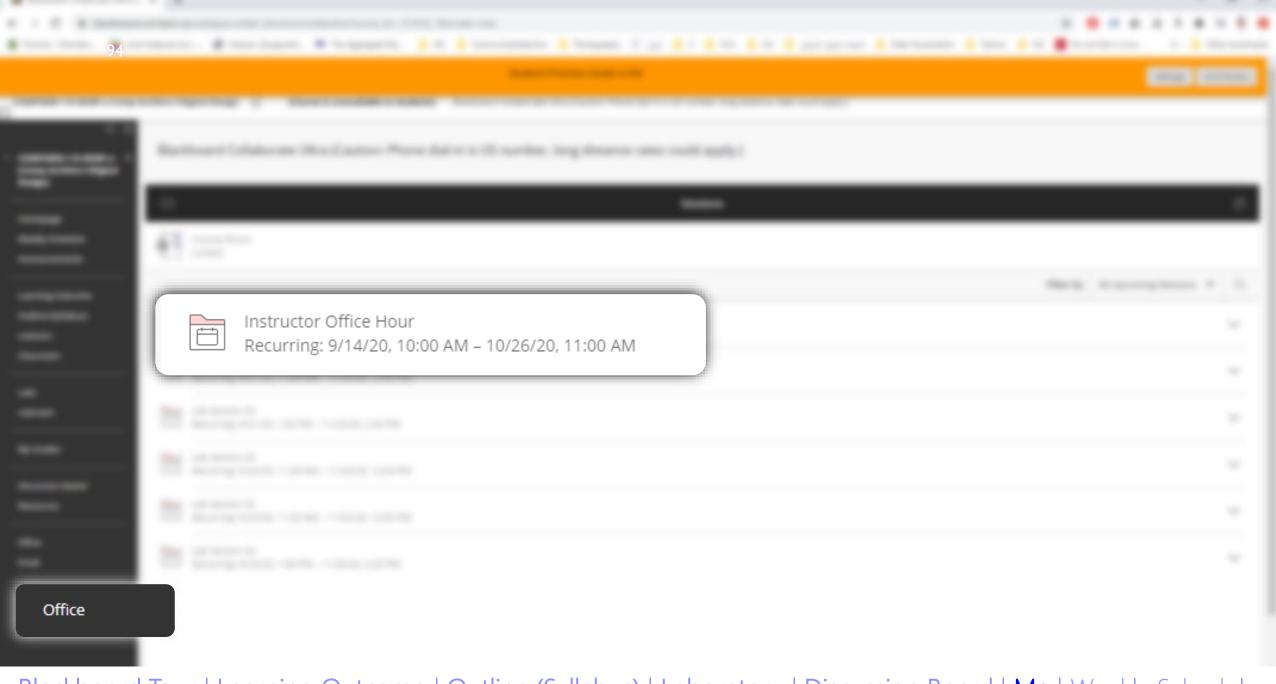
WSDM 2016, 2nd Year PhD, Baker Beach, San Francisco, USA.

Hossein Fani, PhD
Assistant Professor, School of Computer Science, Faculty of Science, University of Windsor
Room 5111, Lambton Tower
hfani@uwindsor.ca; hossein.fani@uwindsor.ca
hfani.myweb.cs.uwindsor.ca

OFFICE

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
  |---> Office

Instructor Office Hour
Recurring: 9/14/20, 10:00 AM – 10/26/20, 11:00 AM

Office

WEEKLY SCHEDULE

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
   |---> Weekly Schedule
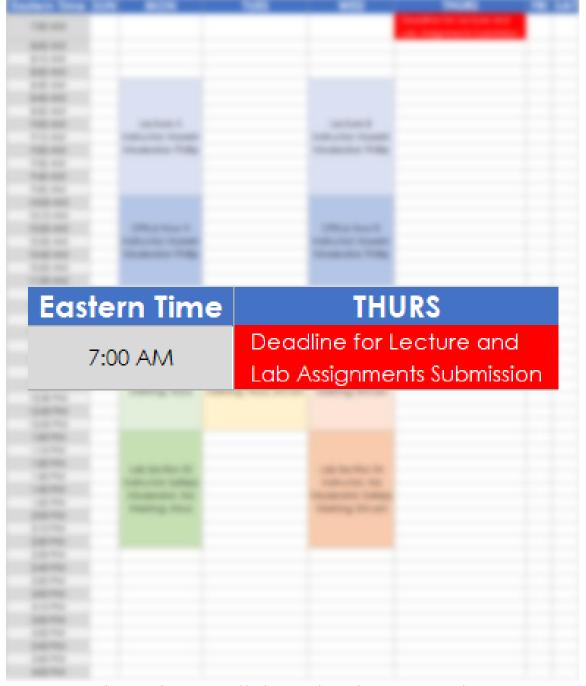
Eastern Time	SUN	MON	TUES	WED	THURS	FRI	SAT
7:00 AM					Deadline for Lecture and Lab Assignments Submission		
8:00 AM							
8:10 AM							
8:20 AM							
8:30 AM							
8:40 AM							
8:50 AM							
9:00 AM		Lecture A Instructor: Hossein Moderator: Phillip		Lecture B Instructor: Hossein Moderator: Phillip			
9:10 AM							
9:20 AM							
9:30 AM							
9:40 AM							
9:50 AM							
10:00 AM							
10:10 AM							
10:20 AM		Office Hour A Instructor: Hossein Moderator: Phillip		Office Hour B Instructor: Hossein Moderator: Phillip			
10:30 AM							
10:40 AM							
10:50 AM							
11:00 AM							
11:10 AM							
11:20 AM							
11:30 AM		Lab Section 51 Instructor: Saiteja Moderator: Ala Marking: Ariya	Lab Section 55 Instructor: Saiteja, Ala Moderator: Ala, Saiteja Marking: Ariya, Shivani	Lab Section 52 Instructor: Ala Moderator: Saiteja Marking: Shivani			
11:40 AM							
11:50 AM							
12:00 PM							
12:10 PM							
12:20 PM							
12:30 PM							
12:40 PM							
12:50 PM							
1:00 PM		Lab Section 53 Instructor: Saiteja Moderator: Ala Marking: Ariya		Lab Section 54 Instructor: Ala Moderator: Saiteja Marking: Shivani			
1:10 PM							
1:20 PM							
1:30 PM							
1:40 PM							
1:50 PM							
2:00 PM							
2:10 PM							
2:20 PM							
2:30 PM							
2:40 PM							
2:50 PM							
3:00 PM							
3:10 PM							
3:20 PM							
3:30 PM							
3:40 PM							
3:50 PM							
4:00 PM							

Blackboard Tour | Learning Outcome | Outline (Syllabus) | Laboratory | Discussion Board | Me | Weekly Schedule

Eastern Time	SUN	MON	TUES	WED
8:30 AM		Lecture A Instructor: Hossein Moderator: Phillip		Lecture B Instructor: Hossein Moderator: Phillip
8:40 AM				
8:50 AM				
9:00 AM				
9:10 AM				
9:20 AM				
9:30 AM				
9:40 AM				
9:50 AM				
10:00 AM		Office Hour A Instructor: Hossein Moderator: Phillip		Office Hour B Instructor: Hossein Moderator: Phillip
10:10 AM				
10:20 AM				
10:30 AM				
10:40 AM				
10:50 AM				
11:00 AM				

Eastern Time	SUN	MON	TUES	WED
11:30 AM				
11:40 AM		Lab Section 51 Instructor: Saiteja Moderator: Ala Marking: Ariya	Lab Section 55 Instructor: Saiteja, Ala Moderator: Ala, Saiteja Marking: Ariya, Shivani	Lab Section 52 Instructor: Ala Moderator: Saiteja Marking: Shivani
11:50 AM				
12:00 PM				
12:10 PM				
12:20 PM				
12:30 PM				
12:40 PM				
12:50 PM				
1:00 PM		Lab Section 53 Instructor: Saiteja Moderator: Ala Marking: Ariya		Lab Section 54 Instructor: Ala Moderator: Saiteja Marking: Shivani
1:10 PM				
1:20 PM				
1:30 PM				
1:40 PM				
1:50 PM				
2:00 PM				
2:10 PM				
2:20 PM				

Eastern Time	THURS
7:00 AM	Deadline for Lecture and Lab Assignments Submission

CODELINE

|---> COMP2650-1-R-2020F-a Comp Architre I:Digital Design
    |---> Lectures
        |---> W01: Meet & Greet, Number Systems I
            |---> Codeline

THANK YOU