CS1002 Programming Fundamentals (CYSEC)

Fall 2022

23rd August 2022

Jawad Hassan

Lecture 1

National University of Computer and Emerging Sciences, Islamabad

Welcome to Cyber Security

Welcome!

Plan your self for next four years with Cyber Security

Ubiquity of Communication Devices

Figure 1: Computers are everywhere, processing and generating data.

Internet of Things

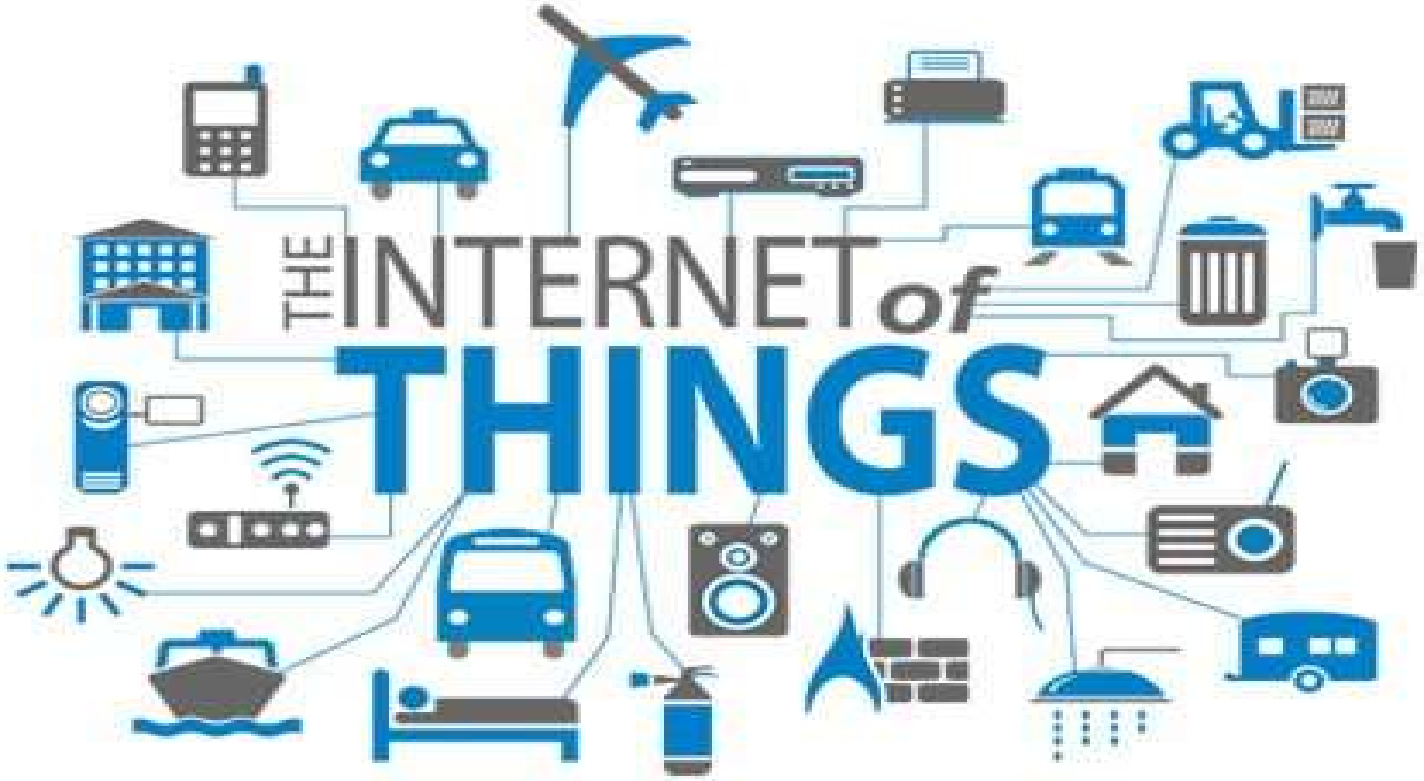

Figure 2: Everything is connected and sharing data ...

What is cyber Security?

Figure 3: What is Cyber Security?

Application of cyber security

Man vs Machine

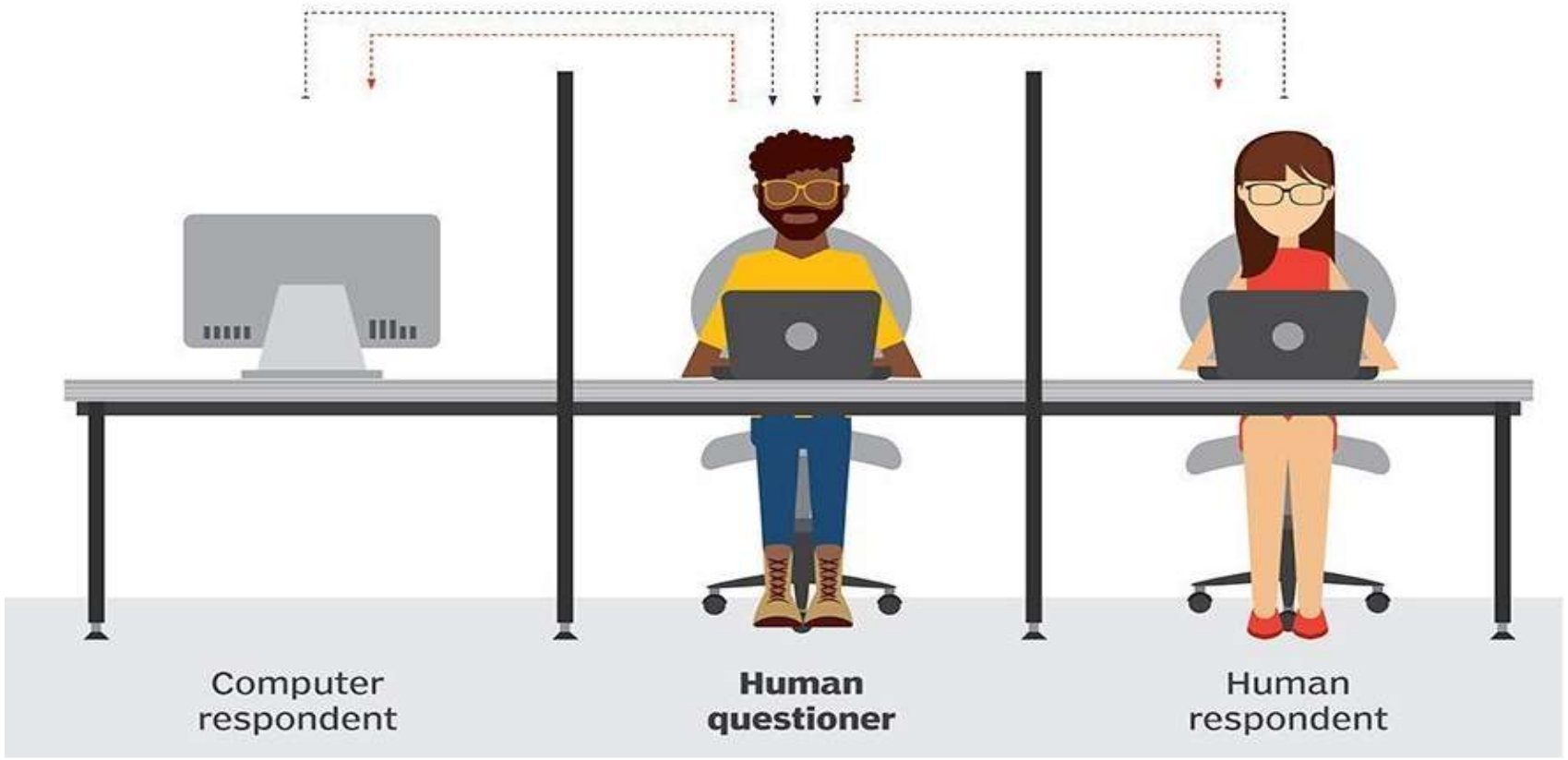

Figure 4: Man vs Machine

Objectives of Programming Fundamentals

- Problem Abstraction
  - Problem understanding from a computational perspective
- Design
  - Developing a conceptual solution
- Implementation
  - Implementing the solution as a computer program
- Testing
  - Validating the implementation

Programming is like Legos

Figure 5: Small number of primitive constructs! [basic tools and pieces]

Programming and Legos!

Figure 6: You can build huge, elaborate structures out of these pieces!

What is the most important skill of a Computer Scientist ?

- What is the most important skill for a computer scientist?

  a) To think like a computer.
  b) To be able to write code really well.
  c) To be able to solve problems.
  d) To be really good at math.

Why Google and why not Bing ?

Skills of a Computer Scientist

- A computer scientist must have these three Critical Skills:
  1. Problem Solving
  2. Programming (master of communicating his solutions to computer)
  3. Strong Math/Physics Knowledge

Our Objectives

- To teach you problem solving capabilities by teaching you how to think like a computer scientist.
- This involves teaching the complete steps from problem understanding to testing your solution.

How you can become a good CS ?

- No Royal Way.
- Dedication / Ambition
- Hard Work
- Punctuality
- Professionalism (Integrity and Honesty).

- **Simple Rule: Program 2-3 Hours a Day to Pass it [Learning]**

What Subjects to Specially Focus on in CS?

- Problem Solving
- Programming
- Maths (every single subject).

Skills Required for an Excellent Computer Scientist

- Computational Thinker
- A Keen Mathematics (Linear Algebra, Probability and Calculus are extremelly important)

What to do for being Successful in CS1002 Or In Semester System

- Be punctual
  - Do not put off your today work till tomorrow.
  - Start on the given task as soon as it is given.
  - Set your goals, for each day, week and month.
- Be attentive and proactive in the class.
- Communicate your problems and discuss them immediately. Don't be Shy and ask questions.
- Think big and do not limit yourself to the class content, **explore and imagine.**

Advantages of doing good in CS1002

- Many
  - Will help you to appreciate your own mind power and act as master of computer.
  - Can help you to understand the other CS core courses throughly.
  - You can start working on real problems soon after completing it and CS1002.

I Just Hate Teaching Passive Classes

- Be Proactive
  - Ask questions → Questions are gateway to your understanding
  - Stop me right there when you do not understand anything.
  - The more responsive you will be better you will learn.
- Rules For Class participation
  - No talking and laughing during a lecture
  - If you have any question or a comment/suggestion, raise your hand and only speak if you are given permission to do so.

Class Discipline

- No use of mobile phones during a lecture
  - Mobiles must be switched off at the start of a lecture and must be put in a bag or pocket i.e. they should not be seen in anyone's hand or on desk.
- No attendance for late comers (Be aware Attendence < 80 % == Debarred means no Final Exams)
  - Attendance will be taken at the start of a class only.
  - In the case of leaving during a lecture (due to any emergency), you will tell me so that you can be marked absent. Try to finish all things before coming to a class.
  - Moreover Quizzes will be mostly taken during first 10 minutes, so being late == losing marks as well.
- Lecture Rooms
  - Section-C: Tuesday (C-409 8:30-09:50), Thursday (C-405 8:30-09:50)
  - Section-D Tuesday (B-227 10:00-11:20), Thursday (C-406 10:00-11:20)
  - Labs: Friday (Section-C 8:30-11:50 & Section-D 2:15-5:00)

Credit Hours

- **Total credit hours = 4**
  - **3 credit hours of class lectures**
  - **1 credit hour of Lab (equivalent to 3 hours of Lab work)**

Lab Work

- Lab Instructors are to be consulted for Lab related issues.
- Do not switch lab sections without asking permission from the lab instructors.
- Laboratory attendance will be taken in each lab.
- For queries related to Lab exercises or assignments, consult the lab instructors only.
- The rules related to the labs will be announced in the laboratory session.

Grading Criteria

- Quizzes (5+2) = 10% →~announced or un-announced
- Assignments (5) = 15 % → ~ Home works (Ungraded)
- Project (1) = 10%
- Sessional Exams (2) = 25 % (Paper-based exam)
- Final (1) = 40%

Grading Policy: **Absolute**

Policy about missed assessment items in the course

- **Retake** of **missed assessment items** (other than **midterm/ final exam**) will not be held (**no retake of assignment/quiz/project**).

  Late submission will be accepted (until certain time) with marks deductions.

- For a missed midterm/ final exam, an exam retake/ pretake application along with necessary evidence are required to be submitted to the department secretary. The examination assessment and retake committee decides the exam retake/ pretake cases.

Course Plagiarism Policy

Plagiarism in any assessment item will result **zero marks** in that assessment.

Repeated case of plagiarism will be reported to the disciplinary committee and may result in zero marks **in the whole category**.

If plagiarism is detected, student will have **1 week** from the date of announcement to defend the charges

Detailed Course Contents (1/2)

List of Topics	No. of Weeks	Contact Hours
- Problem-solving, Basic flowchart, block diagram, and programming languages. - Primitive data types, input/output (hello world). - Signed and unsigned data types, constants and variables.	**1**	**3**
- Arithmetic operators (+, -, *, /, % and their compound counterparts) with their associativity and precedence. - Bit wise operators	**2**	**6**
- Function prototypes, definition, and calling.	**1**	**3**
- Conditional/selection structures. - Comparison and logical operators. - if, if. . .else and if else if structure. - Switch statement, *break* statement. - Ternary operator.	**2**	**6**
- Repetition structures. - Pre/post increment/decrement operators. - while loop (sentinels + condition). - Loop with *for.* - Loop with *do-while.* - Nesting of *while*, *for* loop and *continue* statement.	**3**	**9**

Detailed Course Contents (2/2)

- Introduction to Arrays. - Array initialization and representation. - Char arrays. - Multi-Dimensional Arrays (MDA). - MDA representation in memory.	**1.33**	**4**
- Aliases, parameters passing by value and by reference (passing arrays). - Function calling order and stack (function within a function). - Recursion	**1.66**	**5**
- Header files (creating own file). - Files handling - Opening flags (app mode).	**1**	**3**
- Pointers. - const. vs. non-const. pointers, a pointer to const. data vs. a pointer to non-constant data. - Using pointers. - Dynamic memory allocation. - Array of pointers.	**2**	**6**
Total	**15**	**45**

Our Team

- **Course Instructor**
  - Jawad Hassan
    - Section C and D
    - Office: 202-E
    - **Email:** jawad.hassan@nu.edu.pk
- **Lab Instructors**
  - Ms. Sara Afzal (209-D A block)
    - **Email:** sara.afzal@nu.edu.pk
  - Ms. Ayeha Qamar
    - **Email:** ayesha.qamar@nu.edu.pk
- **Course Coordinator**
  - Dr Mudassar Aslam
    - Section A and B
    - Office: 504e (5th Floor, Block C)
    - **Email:** mudassar.aslam@nu.edu.pk

Course Coordination

- **Online course content & coordination**
  - Slate Course folder, flex for attendance and evaluations
  - Google class room (https://classroom.google.com)
  - Join class room by pressing +
  - class code: **liimgp2**

Course Consultation

Consultation Hours:

Schedule and Office Hours (Fall-2022)						
Days	**8:30 AM 09:50 AM**	**10:00 AM 11:20 AM**	**11:30 AM 12:50 AM**	**13:00 AM 14:20 PM**	**14:20 PM 15:50 PM**	**15:55 PM 16:50 PM**
Monday			**Office Hour**	**Office Hour**		
Tuesday	PF(CY-C)	PF(CY-C)		Number Theory		
Wednesday		**Office Hour**	**Office Hour**	**Office Hour**		
Thursday	PF(CY-C)	PF(CY-C)		Number Theory		
Friday	Appointment via Email (**jawad.hassan@nu.edu.pk**)					

Helping hours (HH)

Course Books

- Text Books
  - Starting Out With CPP (8th or 9th Edition) By Tonny Gaddis (Locally Available)
  - How to think like Computer Scientist, Interactive Edition [Freely available online; Excellent Online Resource for Interactive Learning]
- Reference Books
  - C++ How to Program by Deitel & Deitel (8th Edition)
  - C++ Without Fear : A Beginner's Guide that Makes you Feel Smart (English) 2nd Edition
  - Accelerated C++.
  - C++ Premier, 5th Edition

Softwares

- Ubuntu (or any other recent Linux version) as operating system.
  - I would recommend installing Ubuntu LTS 18.04 or 19.04
    - Make copy of your important data before installation.
    - Read the instructions for steps to perform after the installation
- GNU compiler suite

Live Programming & Visualization Environments

- Blockly Games, https://blockly-games.appspot.com/ or Snap http://snap.berkeley.edu/snapsource/snap.html
- C++ Live Program Visualization
  - http://cpp.sh/
  - https://www.jdoodle.com/online-compiler-c++
  - https://www.tutorialspoint.com/compile_cpp_online.php
  - http://rextester.com/l/cpp_online_compiler_visual

Tips and Advices

- **Self belief**
- **Honesty**
- **Respect**
- **Annual vs Semester System**
- **Attendance**
- **Deadlines**
- **Be interactive**
- **Make good friends**
- **Time table**

Assignment 0 (No credit assignment)

- Task 1
  - Install Ubuntu on your machine or in a VM
  - Download Desktop Ver 22.04 LTS or any other LTS version (https://ubuntu.com/blog/tag/LTS)
  - Install GNU Compiler Collection (GCC) on Ubuntu
- Task 2
  - Time table of semester
- Task 3
  - Join Google Class Room
  - class code: liimgp2

Play Some Games

- https://blockly-games.appspot.com/
- https://scratch.mit.edu/projects/editor/?tutorial=getStarted

QUESTIONS

THANK YOU ALL