

$$F = G \frac{m_1 m_2}{d^2}$$

# Design and Analysis of Algorithms

Bilal Khalid Dar

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$$

$$\frac{df}{dt} = \lim_{h \rightarrow 0} \frac{f(t+h) - f(t)}{h}$$

$$\phi(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$dS \geq 0$$


$$F - E + V = 2$$

$$E = mc^2$$

# About me

<b><u>Name:</u></b>	Bilal Khalid Dar PHD – Software Engineering (in progress)
<b><u>Qualification:</u></b>	MS – Software Engineering (Gold Medalist) BS – Software Engineering (Silver Medalist) (COMSATS Islamabad)
<b><u>Distinctions:</u></b>	ExcITe Cup 2016 FITT Conference'16 Open House 2016
<b><u>Prior Experience:</u></b>	FAST (Lab Instructor) COMSATS Islamabad (Teaching Associate) SZABIST Islamabad (Lab Engineer),
<b><u>Current Responsibilities</u></b>	<ol style="list-style-type: none"><li>1. Lecturer</li><li>2. Head TA Hiring Committee</li><li>3. Head/Mentor FAST Gaming Club</li><li>4. Researcher at Software Engineering and Automation Lab</li></ol>
<b><u>Office:</u></b>	Room 502 G (C Building)
<b><u>Email:</u></b>	bilal.khalid@nu.edu.pk

# Office Hours

 <b>Schedule and Office Hours (Spring-2022)</b>						
<b>Days</b>	8:30 AM 09:50 AM	10:00 AM 11:20w AM	11:30 AM 12:50 AM	01:00 PM 02:20 PM	02:30 PM 03:50 PM	3:55 PM 5:15 PM
<b>Monday</b>			Algo S A310		Algo R C408	SQE P C406
<b>Tuesday</b>			<b>Office Hours</b>		<b>Office Hours</b>	SQE R A311
<b>Wednesday</b>			Algo S A311		Algo R C407	SQE P C404
<b>Thursday</b>			<b>Office Hours</b>		<b><u>Office Hours</u></b>	SQE R C401
<b>Friday</b>			FYP and Research Meetings		FYP and Research Meetings	FYP and Research Meetings

# Your Introduction!

---

Your name

---

How you would rate your coding skills out of 10

---

Your hobby or skills or achievements





# LMS - Google Classroom

Invite sent via Email !

If you have not received any email, please  
drop a mail at [bilal.khalid@nu.edu.pk](mailto:bilal.khalid@nu.edu.pk)



# Course Rules

- Assignments will be submitted handwritten
- **Plagiarism Policy: F in course**
- All quizzes will be both announced and unannounced
- Assignment assessment is based on the quizzes (60/40 ratio). In case of missed quiz for the assignment, 40% marks will be deducted from the assignment (if it is not plagiarized)
- **Missed or late assignment will be marked zero**
- No make up assignments or quizzes
- 3 days for query after every assignment/quizzes



# Class Rules

- You can be late no more than 10 minutes (else you are absent)
- Bring a pen and a register/copy/paper
- Mobile phones shall be OFF/SILENT (RS 2000/- Fine)
- Do not wander around in Class

# Marks Distribution

Assessment Item	Number	Weight (%)
Quiz	8	10
Project	1	10
Midterm Exam 1	1	15
Midterm Exam 2	1	15
Assignments	4	10
Final Exam	1	40



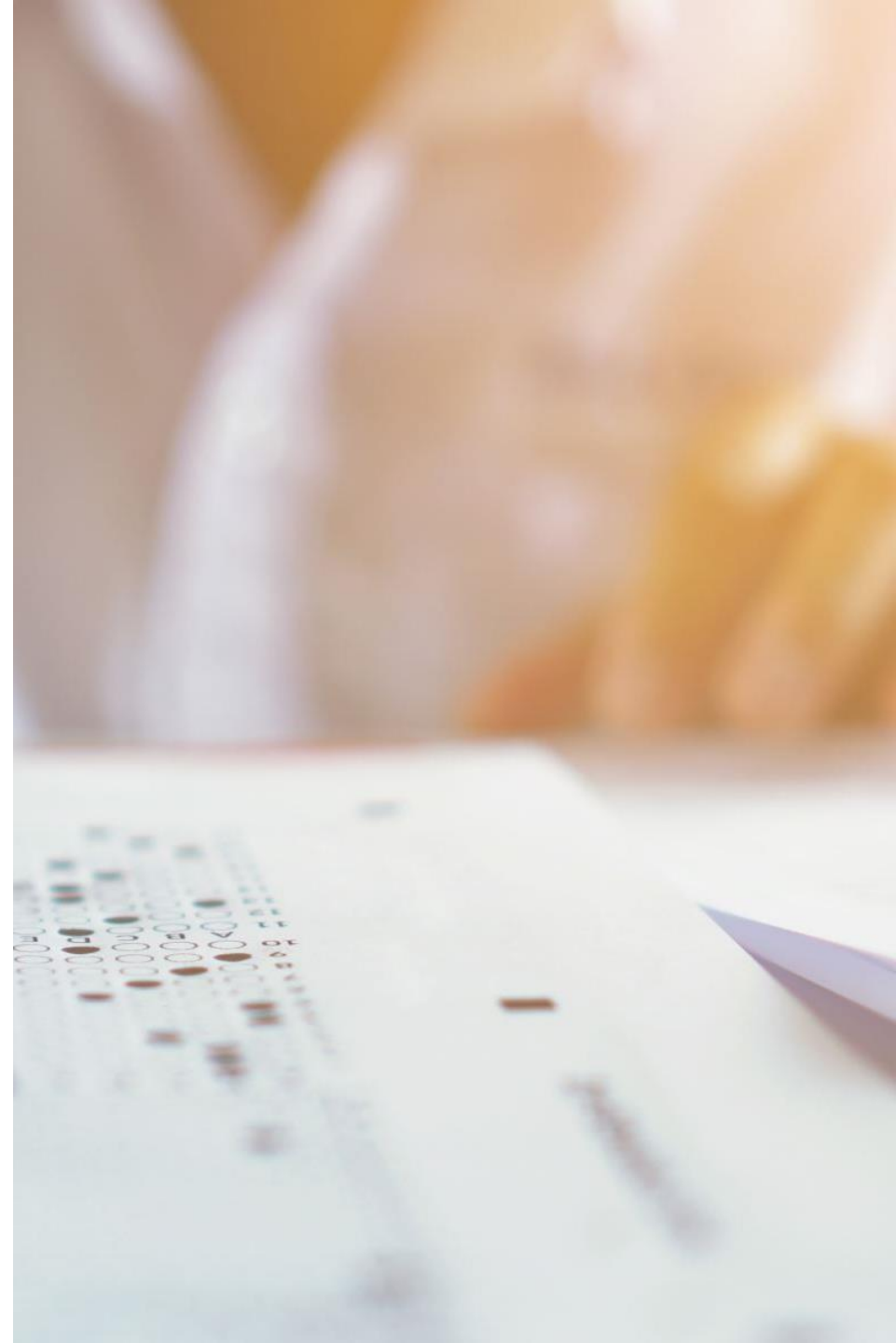
Absolute Grading Scheme: Total Marks [%] Grade	
$\geq 90$ A+	66-69 C+
86-89 A	62-65 C
82-85 A-	58-61 C-
78-81 B+	54-57 D+
74-77 B	50-53 D
70-73 B-	$\leq 49$ F



# NOTE

---

- Grading will be individual even for group tasks. So better to shine with your own work in hand otherwise don't complaint





## Formula for GOOD GRADES

All of you have **A+**

Just do the following

- Maintain Attendance
- Positive class attitude and Participation
- Perform all class task
- Do Quizzes and Assignments YOURSELF
- Follow all Rules!

And keep your **A+**

# Textbook and reference material

---



Introduction to Algorithms (Text Book) Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein Third Edition, MIT Press



Any web material (consult authentic material e.g. on some university's website)

Let's start





# What is algorithm

- Algorithm
  - An algorithm is a well-defined and effective sequence of computation steps that takes some value, or set of values, as input and produces some value, or set of values, as output.

# Example



ASUS TUF Dash 15 (2022) Gaming Laptop, 15.6" 144Hz FHD Display, Intel Core i7-12650H, GeForce RTX 3060, 16GB DDR5, 512GB SSD,...

★★★★★ ~ 1,307

**\$1,247<sup>29</sup>**

Ships to Pakistan

More Buying Choices  
\$1,230.08 (12 new offers)

Price may vary by color

Featured

Price: Low to High

Price: High to Low

Avg. Customer Review

Newest Arrivals



SAMSUNG Galaxy Tab A7 Lite 8.7" 32GB Android Tablet w/ Compact, Slim Design, Sturdy Metal Frame, Long Lasting Battery, Gray

★★★★★ ~ 8,294

**-25% \$119<sup>99</sup>** ~~\$159.99~~

Ships to Pakistan

More Buying Choices  
\$115.00 (22 used & new offers)

Price may vary by color

---


Why is the study of algorithms worthwhile?

Some  
questions

---

What is the role of algorithms relative to other technologies used in computers?

---



## Example: sorting

- Input: A sequence of  $n$  numbers  $\langle a_1, a_2, a_3 \dots a_n \rangle$
- Output: A permutation (re-ordering)  $\langle b_1, b_2, b_3 \dots b_n \rangle$  of the input sequence such that  $b_1 < b_2 < b_3 \dots < b_n$

# Correctness of an algorithm

- An algorithm is said to be correct if, for every input instance, it halts with the correct output.
- An incorrect algorithm might not halt at all on some input instances, or It might halt with an answer other than desired on






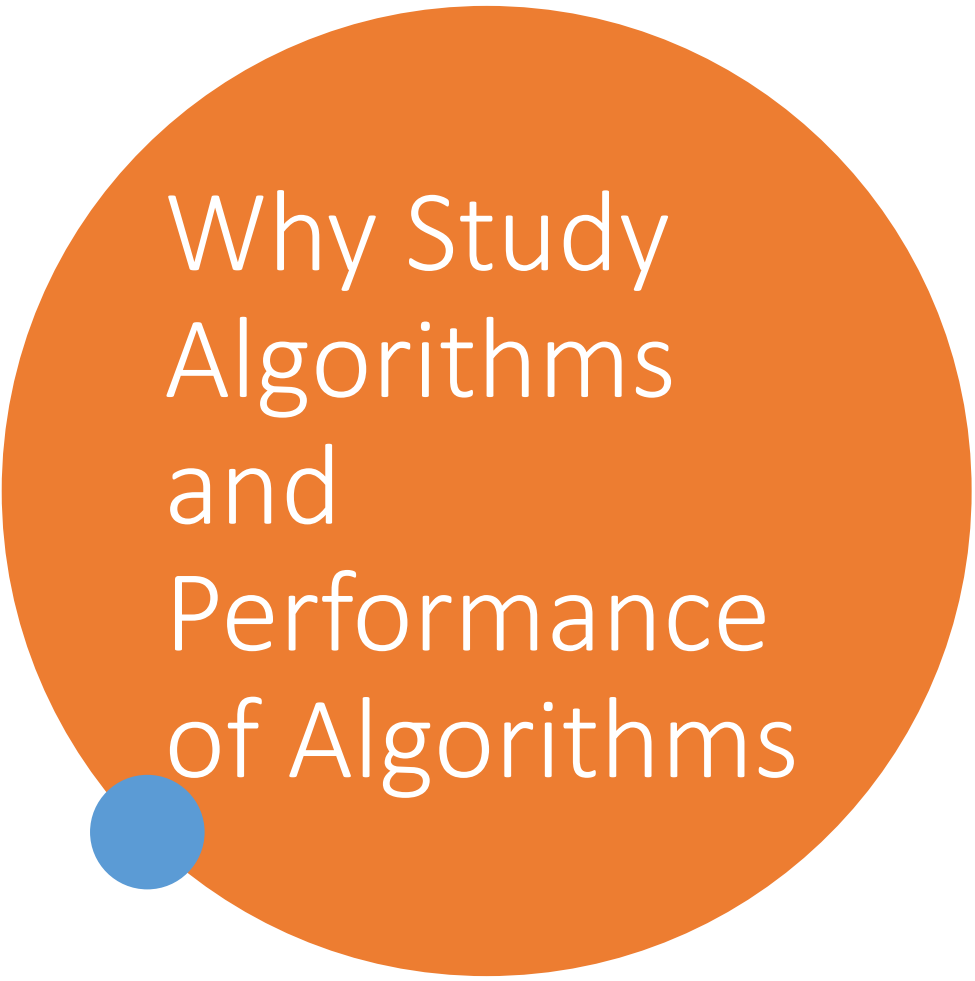
# Problems solved by algorithms

- Sorting/searching are by no mean the only computational problem for which algorithms have been developed.
- Otherwise, we wouldn't have the whole course on this topic
- Practical application of algorithms are ubiquitous and include the following examples
  - Internet world
  - Electronic commerce
  - Manufacturing and other commercial settings
  - Shortest path
  - Matrices multiplication order
  - DNA sequence matching


A large orange circle on the left side of the slide, partially cut off by the edge.

## Common about algorithms

- There are many candidate solutions, most of which are not what we want, finding one that we do want can present quite a challenge.
  - There are practical applications (its not just mathematical exercises to develop algorithms.)
- 
- A series of four yellow curved dashes in the bottom right corner, arranged in a roughly diagonal line from bottom-left to top-right.



# Why Study Algorithms and Performance of Algorithms

- 
- Algorithms help us to understand scalability.
  - Performance often draws the line between what is feasible and what is impossible.
  - Algorithmic mathematics provides a language for talking about program behavior.
  - Performance is the currency of computing.
  - The lessons of program performance generalize to other computing resources.
  - Speed is fun!

# What is Abstract Data Type

- A definition for a data type solely in terms of
  - a set of values and a
  - set of operations on that data type.
- The definition consists of:
- storage structures (data structures) to store the data items and algorithms for the basic operations



# What is a Data Structure

- A data structure is a way to store and organize data in order to facilitate access and modifications.
- No single data structure works well for all purposes
- Need to know the strengths and limitations of several of them.





# Technique

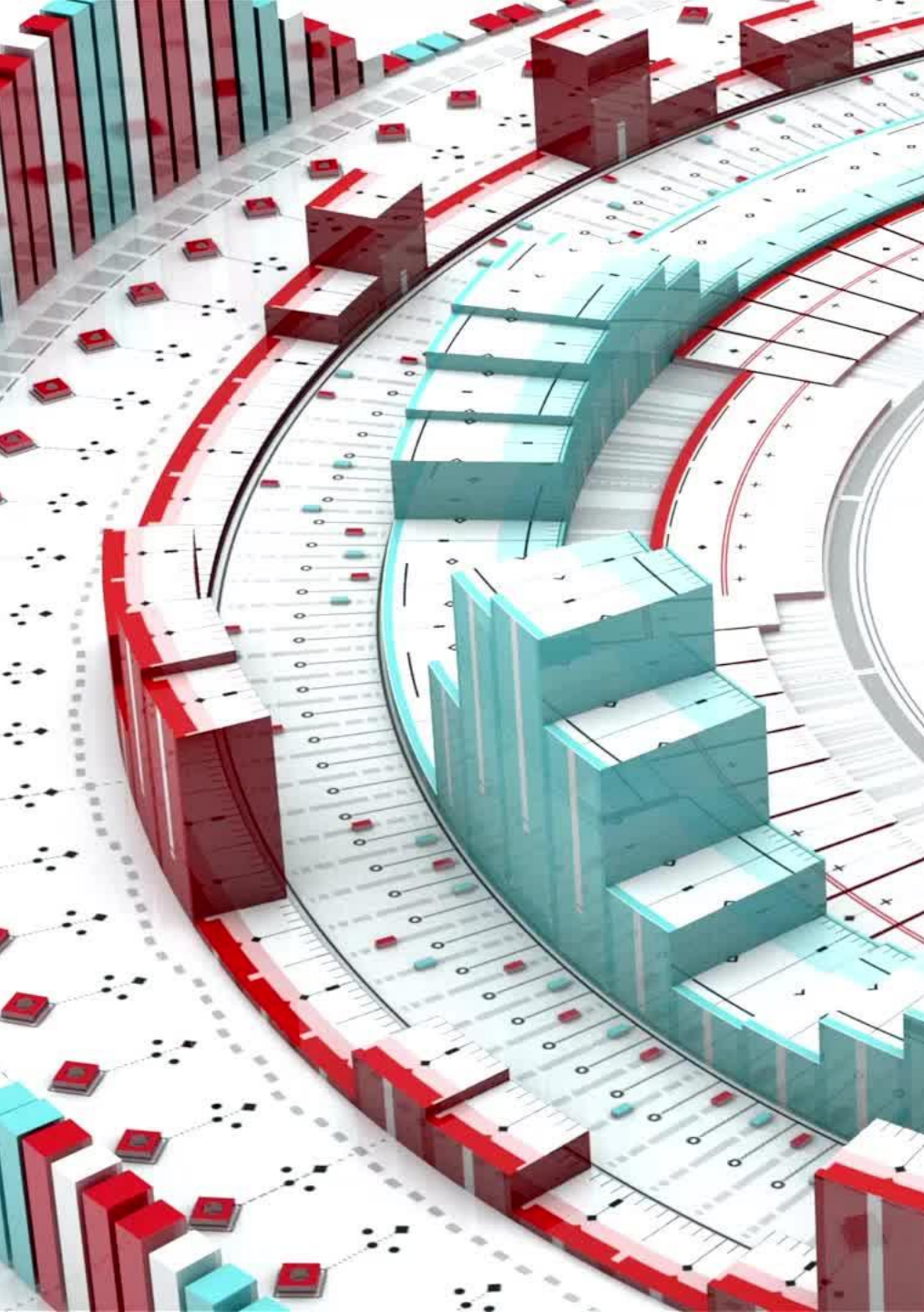
- Can't get a "cookbook" for algorithms?
- Many problems you will encounter don't have any published algorithm.
- So need to learn "techniques" of algorithms design and analysis
- So you develop algorithms in your own, show that they give correct answer and understand their efficiency.
- We will learn several such techniques in later part of this course.



# Algorithms and other technologies

- Is an algorithm a technology like hardware, etc?
- Total system performance depends on choosing “efficient” algorithms as much as choosing fast hardware.





# Algorithms and other advanced technologies

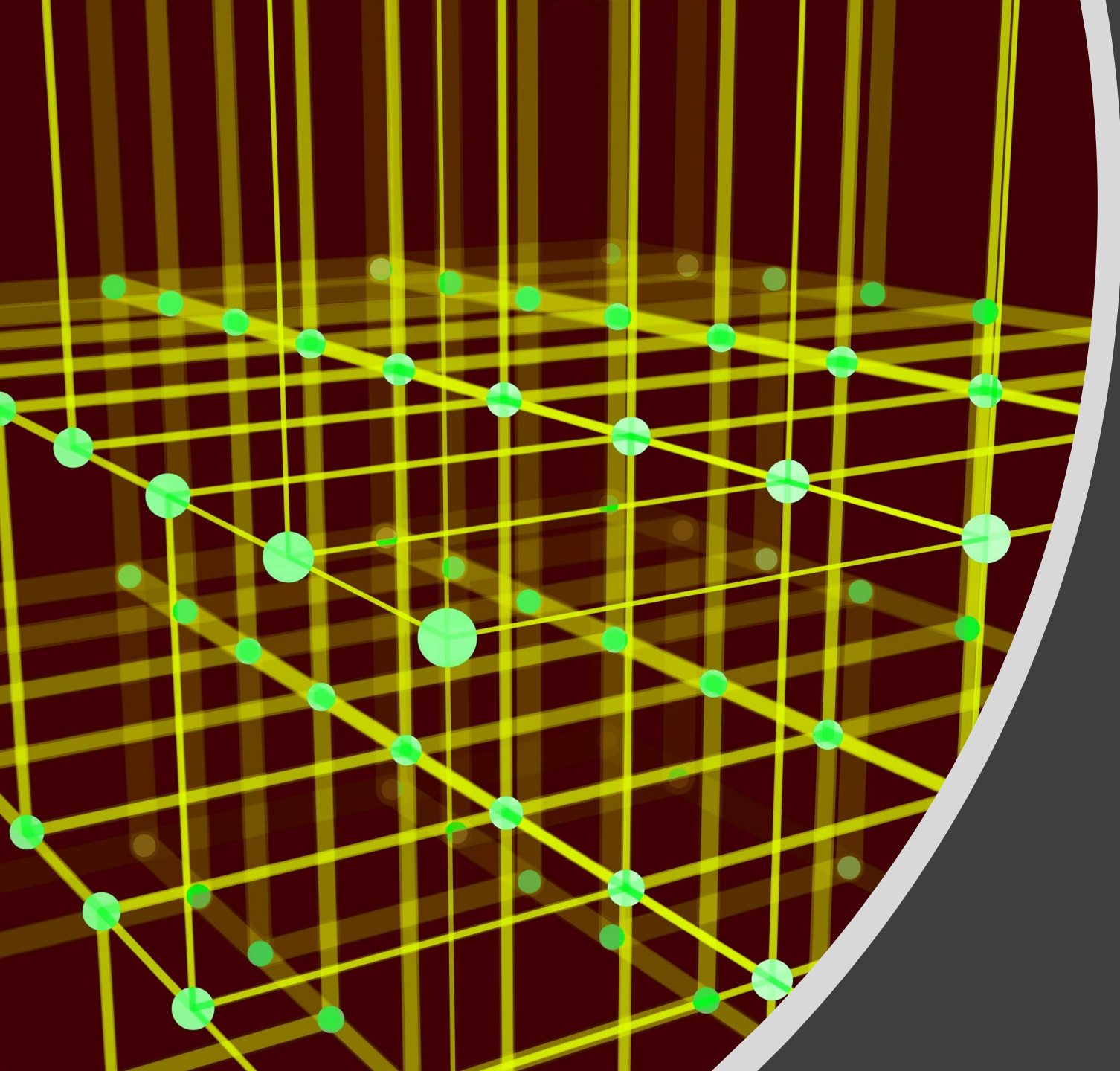
- Hardware with high clock rates, pipelining and superscalar architecture.
- Easy to use graphical user interface (GUI's)
- Object oriented systems.
- Local-area and wide-area networking.
- Are algorithms as important as above technologies?



# Course-Overview

- Basics of Algorithms
- Asymptotic Notations
- Sorting Algorithms
- Divide and Conquer
- String Matching Algorithms
- Graph Theory
- Dynamic Programming





Pseudocode



# Pseudocode

- Pseudocode is a compact and informal high-level description of a program using the conventions of a programming language, but intended more for humans.
- Pseudocode is not an actual programming language.
- So it cannot be compiled into an executable program.



# Pseudocode



- **INPUT** – indicates a user will be inputting something
- **OUTPUT** – indicates that an output will appear on the screen
- **WHILE** – a loop (iteration that has a condition at the beginning)
- **FOR** – a counting loop (iteration)
- **REPEAT – UNTIL** – a loop (iteration) that has a condition at the end
- **IF – THEN – ELSE** – a decision (selection) in which a choice is made any instructions that occur inside a
- selection or iteration are usually indented

# Pseudocode Example

- Calculate Area and Perimeter of Rectangle

```
BEGIN  
NUMBER b1,b2,area,perimeter  
INPUT b1  
INPUT b2  
area=b1*b2  
perimeter=2*(b1+b2)  
OUTPUT area  
OUTPUT perimeter  
END
```



## Pseudocode If Else Example

```
1  
2 BEGIN  
3 NUMBER age  
4  
5 INPUT "Enter your age for driving licence"  
6 OUTPUT age  
7  
8 IF age >= 18 THEN  
9     OUTPUT "You can take driving licence"  
10 ELSE  
11     OUTPUT "You can't take driving licence"  
12 ENDIF  
13 END
```

Pseudocode  
For Loop  
Example

BEGIN

NUMBER counter

FOR counter = 1 TO 100 STEP 1 DO

    OUTPUT counter

ENDFOR

END

Read 10  
numbers and  
find sum of  
even numbers

```
1
2 BEGIN
3 NUMBER counter, sum=0, num
4 FOR counter=1 TO 10 STEP 1 DO
5     OUTPUT "Enter a Number"
6     INPUT num
7     IF num % 2 == 0 THEN
8         sum=sum+num
9     ENDIF
10 ENDFOR
11 OUTPUT sum
12
13 END
```

Find the sum  
of all  
elements of  
array

```
PROCEDURE SUM_Array  
BEGIN
```

```
Input: List numbers[1..n], n
```

```
Output: Sum of an array
```

```
sum=0
```

```
FOR i=0 to n-1
```

```
    sum = sum + numbers[i]
```

```
ENDFOR
```

```
OUTPUT "Sum of numbers in the array"+sum
```

```
END
```





# Reference

Chapter#1 (The Role of Algorithms in Computing)  
Introduction to Algorithms Thomas H.Corman

Thank you

