

# UNIVERSITY OF CENTRAL PUNJAB



SPRING 2025

**Course Title:** Theory of Automata

**Course Code:** CSAL3253-S25-BS-CS-F22-F7

## Assignment No. 04

Course Instructor: Ms. Zar Bakht Imtiaz		
Section: F7	Program: BSCS	Date: 22/06/2025
Submission Date: 29/06/2025	Maximum Marks: 50	
Program Objective:	Course Objective: CO3	Course Learning Objective: CLO3
<b>TO BE FILLED IN BY THE STUDENT</b>		
Student Name:	Registration No:	Sr. No:

### Instructions:

1. No submission after deadline.
2. Assignment must be submitted individually.
3. You will get Zero marks if found any type of cheating.
4. Understanding of the problems is part of the assignments. Answer all questions clearly and concisely.
5. Upload the solved assignment (soft copy) at [university portal](#) before the deadline

### Assignment Topic & Details:

Pushdown Automata

### Q1. DNA Pair Matching Validator

DNA strands use base pairing rules:

- A pairs with T
- C pairs with G

A valid DNA structure is one where the first half of the strand can be reversed and complemented to form the second half.

Valid: ACGT, CGCG, ATGCAT

Invalid: AACG, GTAC, AAGT

Design a PDA to validate such reverse-complement symmetry.

- Use stack to store and compare symbols.
- Describe working for input: ACGT.

## Q2. Browser History Tracker

Your browser app tracks navigation:

- `visit` = user visits new site
- `back` = goes to previous page
- `forward` = goes to forward page

Rule: You must `visit` before using `back` or `forward`, and only go forward if a `back` was previously done.

Model this system using PDA where:

- Stack simulates backtracking.
- Ensure `forward` only after `back`.

## Q3. Directory Navigation System

In a file system, users navigate using:

- `cd` to enter a directory
- `back` to go up one level

Example: `cd cd back cd back back` is valid. `cd back back` is invalid.

Build a PDA to validate proper navigation structure.

- Stack tracks `cd` commands.
- Every `back` must match a `cd`.
- Accept only if stack is empty at the end.

## Q4. Palindrome Checker

You are developing a feature for a language learning app that detects **palindromic sequences** in user-typed binary strings. A string is a palindrome if it reads the same forward and backward (e.g., 0110, 10101).

**Task:**

Design a PDA that accepts all binary palindromes over  $\Sigma = \{0,1\}$ .

- Show how the PDA uses **non-determinism** and the stack to guess the midpoint and verify the reverse half.
- Provide a string parsing on input 10101.