



University of Colombo School of Computing

SCS 1308 - Foundations of Algorithms

LabSheet - 01

Read the following Streamlining Operations at a UCSC Job Fair and answer the questions below.

The UCSC Job Fair is an annual event where students and companies meet for internships and job opportunities. The event involves:

1. **Company Booths:** Students search for booths based on company names, sectors, or booth numbers.
2. **Student Registrations:** Students register upon arrival, and registrations must be processed in the order of their arrival.
3. **Interview Scheduling:** Priority is given to students with higher GPAs, ensuring fairness and efficiency in allocating time slots.

The organizers need a system to manage these operations efficiently using searching and sorting algorithms. Each task will be solved programmatically, focusing on Linear Search, Binary Search, Jump Search, Interpolation Search, Shell Sort, and Radix Sort.

- O1.** A student wants to find a company booth by its name in an unsorted list of booth names.

Write a C program that

1. Accepts an unsorted array of company names.
2. Prompts the user to input a company name to search.
3. Uses Linear Search to find the booth and returns the position if found.

Inputs:

Company Names: {"Tech Innovators", "Code Haven", "AI Solutions", "DataCorp"}

Search for: "Code Haven"

Expected Output:

"Code Haven found at position 1."

02. A student wants to find a company from a sorted list of sectors (e.g., "AI/ML", "Cybersecurity").

Write a C program that

1. Accepts a sorted array of company sectors.
2. Prompts the user to input a sector to search.
3. Uses Binary Search to locate the sector and returns its position if found.

Input:

Sectors: {"AI/ML", "Cloud Computing", "Cybersecurity", "Data Science"}

Search for: "Cybersecurity"

Expected Output:

"Cybersecurity found at position 2."

03. A student wants to locate booth #45 in a sorted list of booth numbers. Write a C program that

1. Accepts a sorted array of booth numbers.
2. Prompts the user to input a booth number to search.
3. Uses Jump Search to locate the booth number and returns its position if found.

Input:

Booth Numbers: {10, 20, 30, 40, 45, 50, 60}

Search for: 45

Expected Output:

"Booth number 45 found at position 4."

04. Sort student registrations by their arrival time for efficient processing. Write a C program that

1. Accepts an array of student arrival times.
2. Sorts the array using Shell Sort.
3. Outputs the sorted arrival times.

Input:

Arrival Times: {10:15, 09:30, 11:00, 10:00, 09:45}

Expected Output:

"Sorted Arrival Times: 09:30, 09:45, 10:00, 10:15, 11:00."

05. Organizers need to prioritize students for interviews based on their GPA. Write a C program that:

1. Accepts an array of GPAs.
2. Sorts the GPAs using Radix Sort.
3. Outputs the sorted GPAs.

Input:

GPAs: {3.5, 4.0, 3.8, 3.2, 3.9}

Expected Output:

"Sorted GPAs: 3.2, 3.5, 3.8, 3.9, 4.0."
