# SDN Controller Placement Optimization Mini-Project

## Authors and Contributions

- **Student 1**: [Name] - Student Number: [Number] - Contribution: [X]%
- **Student 2**: [Name] - Student Number: [Number] - Contribution: [Y]%

*Brief description of how work was divided between team members*

---

## Abstract

*Concise summary of the optimization problem, methods implemented (GRASP, GA, ILP), key results, and main conclusions*

---

## 1. Introduction

### 1.1 Problem Description

- Software Defined Network (SDN) controller placement problem
- Graph G = (N, A) with 200 nodes and 250 links
- Objective: Select n=12 switches to minimize average shortest path length from each switch to its closest SDN controller
- Constraint: Maximum shortest path length between any pair of controllers ≤ Cmax = 1000

### 1.2 Problem Formulation

- Mathematical formulation of the optimization problem
- Decision variables, objective function, and constraints
- Problem complexity and classification

### 1.3 Methodology Overview

- Brief introduction to the three approaches: GRASP, Genetic Algorithm, and Integer Linear Programming

---

## 2. Integer Linear Programming (ILP) Approach

### 2.1 Mathematical Model

- Complete ILP formulation with variables, objective function, and constraints
- Explanation of constraint modeling for the maximum distance requirement

### 2.2 Implementation Strategy

- MATLAB code structure for generating the LP file
- Integration with lpsolve solver
- Parameter settings and solver configuration

### 2.3 MATLAB Code Implementation

```matlab
% Include and explain the MATLAB code that generates the LP file
% Code should be well-commented and structured
```

### 2.4 Results and Analysis

- Solution found by lpsolve within 5-minute time limit
- Objective value achieved
- Running time analysis
- Solution quality assessment

---

## 3. GRASP (Greedy Randomized Adaptive Search Procedure)

### 3.1 Algorithm Design

- Construction phase: greedy randomized selection strategy
- Local search phase: neighborhood structures and improvement procedures
- Overall GRASP framework

### 3.2 Implementation Details

- Candidate list construction and selection criteria
- α parameter for controlling randomization level
- Local search operators and strategies
- Stopping criteria based on runtime limit

### 3.3 Parameter Tuning and Best Settings

- Testing methodology for parameter selection
- Analysis of different α values
- Local search strategy comparison
- Justification for chosen parameter settings

### 3.4 MATLAB Code Implementation

```matlab
% Include and explain the GRASP MATLAB implementation
% Focus on key components: construction, local search, main loop
```

### 3.5 Advanced Strategies Beyond Standard GRASP

- *[If implemented]* Enhanced features such as:
  - Path relinking
  - Memory mechanisms
  - Adaptive parameter control
  - Hybrid approaches

### 3.6 Experimental Results

- Results from 10 runs with 30-second time limit each
- Minimum, average, and maximum objective values
- Convergence analysis and performance consistency

---

## 4. Genetic Algorithm (GA)

### 4.1 Algorithm Design

- Chromosome representation for controller placement
- Population initialization strategy
- Selection mechanism
- Crossover and mutation operators
- Replacement strategy

### 4.2 Implementation Details

- Population size and generation management
- Fitness function design
- Constraint handling for Cmax requirement
- Stopping criteria implementation

### 4.3 Parameter Tuning and Best Settings

- Testing methodology for GA parameters
- Population size optimization
- Crossover and mutation rate analysis

- Selection pressure evaluation
- Justification for chosen parameter settings

### 4.4 MATLAB Code Implementation

```matlab
% Include and explain the GA MATLAB implementation
% Cover key components: initialization, selection, crossover, mutation
```

### 4.5 Advanced Strategies Beyond Standard GA

- *[If implemented]* Enhanced features such as:
  - Multi-objective optimization
  - Island model or parallel GA
  - Adaptive genetic operators
  - Hybrid local search integration

### 4.6 Experimental Results

- Results from 10 runs with 30-second time limit each
- Minimum, average, and maximum objective values
- Population diversity analysis and convergence behavior

---

## 5. Comparative Analysis

### 5.1 Solution Quality Comparison

- Objective value comparison across all three methods
- Statistical analysis of metaheuristic results
- Gap analysis between heuristic and exact solutions

### 5.2 Runtime Performance Analysis

- Computational efficiency comparison
- Convergence speed analysis
- Scalability considerations

### 5.3 Robustness and Consistency

- Variance analysis for metaheuristic methods
- Reliability assessment across multiple runs
- Performance stability evaluation

### 5.4 Method Characteristics

- Strengths and weaknesses of each approach
- Suitability for different problem scenarios
- Implementation complexity comparison

---

## 6. Results Summary

### 6.1 Best Solutions Found

- Optimal/best controller placement configurations
- Objective values achieved by each method
- Constraint satisfaction verification

### 6.2 Performance Metrics Table

| Method | Min Objective | Avg Objective | Max Objective | Best Runtime | Avg Runtime |
|---|---|---|---|---|---|
| ILP | [value] | N/A | N/A | [time] | [time] |
| GRASP | [value] | [value] | [value] | [time] | [time] |
| GA | [value] | [value] | [value] | [time] | [time] |

### 6.3 Statistical Analysis

- Confidence intervals for metaheuristic results
- Hypothesis testing for method comparison
- Performance distribution analysis

---

## 7. Conclusions and Future Work

### 7.1 Key Findings

- Main conclusions about method performance
- Problem-specific insights
- Practical implications for SDN controller placement

### 7.2 Method Recommendations

- Which method to use in different scenarios
- Trade-offs between solution quality and computational time
- Guidelines for parameter selection

### 7.3 Future Research Directions

- Potential improvements to the implemented methods
- Extensions to larger problem instances
- Integration with real SDN deployment scenarios

---

## References

*Academic references for algorithms, optimization techniques, and SDN-related work*

---

## Appendices

### Appendix A: Complete MATLAB Code Listings

- Full ILP generation code
- Complete GRASP implementation
- Complete GA implementation
- Utility functions and helper code

### Appendix B: Detailed Experimental Results

- Complete results tables from all runs
- Statistical analysis details
- Performance graphs and charts

### Appendix C: Problem Instance Data

- Description of the network topology
- Node and link data structure
- Distance matrix characteristics