Algorithms and Data Structures Using Java

- Soumya

Topics Covered

- Define the Problem
- Identify the problem
- Problem solving basics

What is a Problem?

- A State of difficulty that needs to be resolved.
- Problems exist where goals need to be attained and there is uncertainty about solution.
- A Problem is an opportunity for improvement.
- A Problem is the difference between your current state and your goal state.

Problem Faced in Everyday in Life

➔ **Examples:**

- Should I wear casual or formal today?
- Should I watch TV or go out to cinema?
- what career?
- what course?
- What shoes?
- Everything needs a **DECISION AS A SOLUTION TO THE PROBLEM**.

What happens when bad decisions are made?

WASTAGE OF TIME AND RESOURCES

Six steps to ensure a Best decision

- Identify the problem
- Understand the problem
- Identify alternative ways to solve the problem
- Select the best way to solve the problem from the list of alternative solutions
- List instructions that enable you to solve the problem using selected solution
- Evaluate the solution.

Take the problem of what to do this evening

- How do the individual wish to spend the evening
- knowledge base-playing chess example
- watch television,go to movie etc.,
- select best one
- prepare list of steps that will result in a fun evening.
- are we having the fun yet?

Define the problem

- **Inputs**: The inputs to a problem are the data that is used to solve the problem.

  Example:the inputs to a sorting problem might be a list of unsorted numbers.

- **Outputs**: The outputs of a problem are the results that are produced by the problem solving algorithm.

  Example: the output of a sorting problem might be a list of sorted numbers.

Define the problem

- **Constraints:**

– The constraints of a problem are any restrictions that must be considered when solving the problem.

– Example: a sorting problem might have a constraint that the algorithm must be efficient, meaning that it must be able to sort the list of numbers in a reasonable amount of time.

Define the problem

Here are some tips for defining problems

– Be clear and concise.

– Use unambiguous language.

– Identify all of the inputs, outputs, and constraints of the problem.

– Write the definition in a way that is easy to understand for both humans and computers.

Identify a problem

To identify a problem in problem solving, you can:

– Look for opportunities to improve efficiency or accuracy.

Are there any tasks that are currently being done manually that could be automated?

Are there any calculations that could be made more accurate by using a computer?

– Look for patterns or trends in data.

Can you identify any patterns or trends in your data that could be used to make better predictions or decisions?

Identify a problem

• Look for ways to optimize existing systems.

–Can you find ways to improve the performance or scalability of your existing systems?

• Look for new ways to use data.

– Is there any data that you are not currently using that could be used to solve problems in new ways?

• Talk to stakeholders.

–What are the biggest challenges that your stakeholders are facing? Can you think of any computational solutions to these challenges?

Problem Solving

• Problem solving can be broken down into the following steps:

Problem formulation: The first step is to clearly define the problem that needs to be solved. This involves understanding the inputs and outputs of the problem, as well as the constraints.

Algorithmic design: The next step is to design an algorithm to solve the problem. An algorithm is a step-by-step procedure for solving a problem.

Problem Solving

• **Implementation:**

– The next step is to implement the algorithm using a programming language.

• **Testing:**

– The next step is to test the implementation to ensure that it is correct and efficient.

• **Deployment:**

– Once the implementation has been tested, it can be deployed to production.

Problem Solving

- Here are some tips for effective computational problem solving:
  - Choose the right tools.
  - Break the problem down into smaller pieces.
  - Use abstraction.
  - Test your solutions thoroughly.
  - Document your work.

Computational problem solving

- Computational problem solving is the iterative process of developing computational solutions to problems.
- Computational solutions are expressed as logical sequences of steps (i.e. algorithms), where each step is precisely defined so that it can be expressed in a form that can be executed by a computer.
- Much of the process of computational problem solving is thus oriented towards finding ways to use the power of computers to design new solutions or execute existing solutions more efficiently.

Computational problem solving

- **Decomposition**:In computing, decomposition is the process of breaking down a task into smaller, more-manageable parts.
- **Pattern Recognition**:By identifying patterns, we can create rules and solve
- more-general problems.
- **Abstraction**:he practice of ignoring certain details in order to come up with a solution that works for a more general problem.
- **Algorithm**:An algorithm is a sequence of instructions or a set of rules to get something done.

Problem

Write the steps to find the number is odd or even.