

# **PROGRAMMING LANGUAGE ISSUES**

- **Procedural vs. Nonprocedural**
- **Goals of Software Engineering**
- **Language-Specific Issues**
  - **Control Structures**
  - **Data Typing**
  - **Subprograms and Collections**
  - **Structured Programming**
  - **Object-Oriented Programming**
  - **Application Domains**
- **Compiler-Specific Issues**
- **Organizational Issues**
  - **Culture and Psychological View**
  - **Education and Training, Resources Required, and Cost**
- **Language Selection**
  - **Trends by Application Domain**
  - **Criteria for Selection**
  - **Assessment**

# Software Development Lifecycle

**System Engineering**

**Analysis**

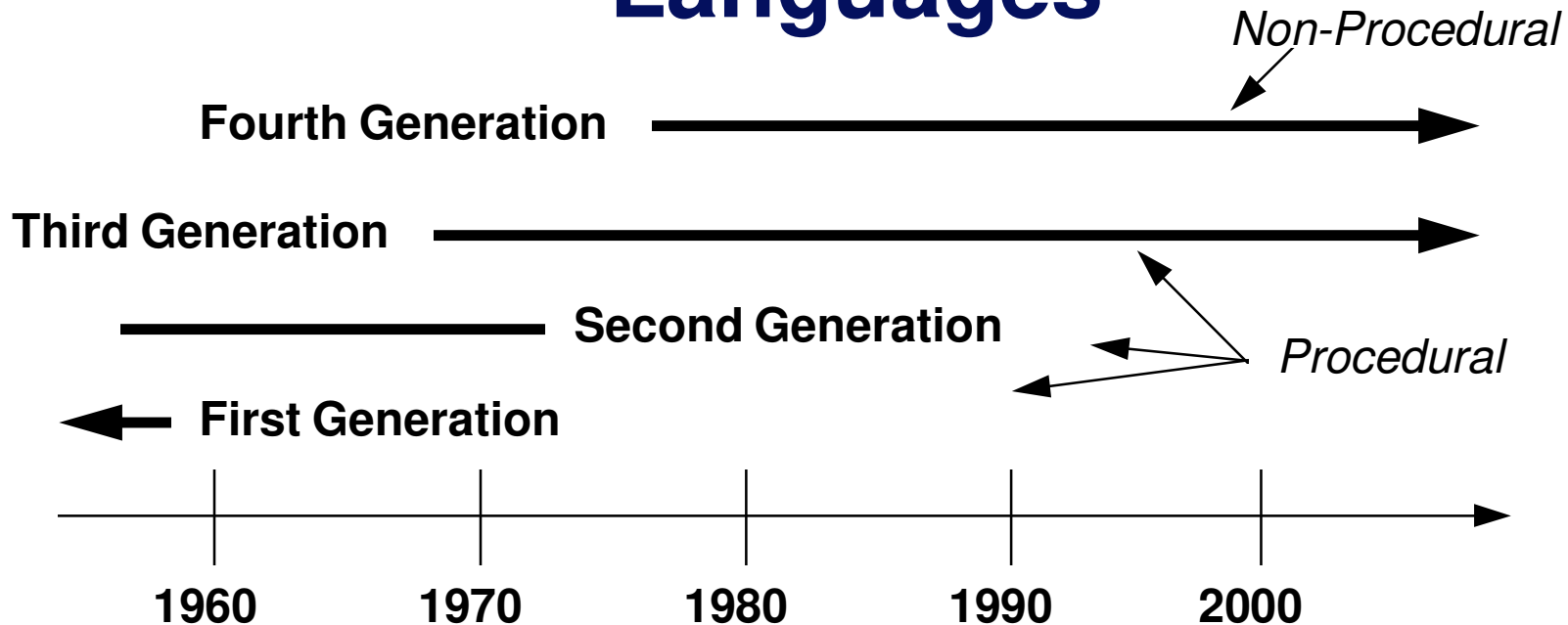
**Design**

**Coding**

**Testing**

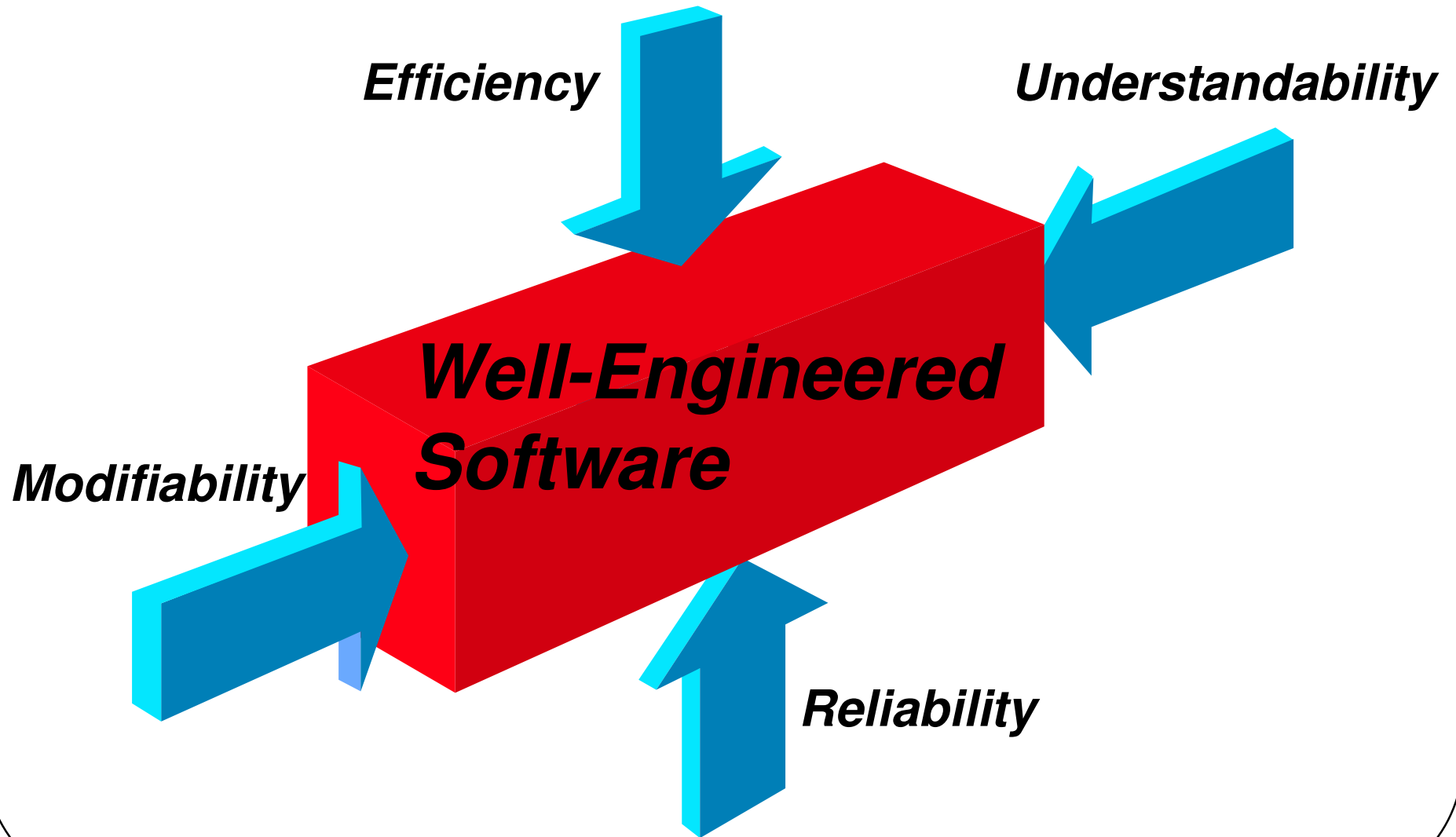
**Maintenance**

# Procedural Vs. Nonprocedural Languages



- **Procedural Language** - Capable of detailing the steps to be taken to achieve desired results
- **Non-Procedural Language** - Capable of detailing the desired results (the language translator creates the steps)

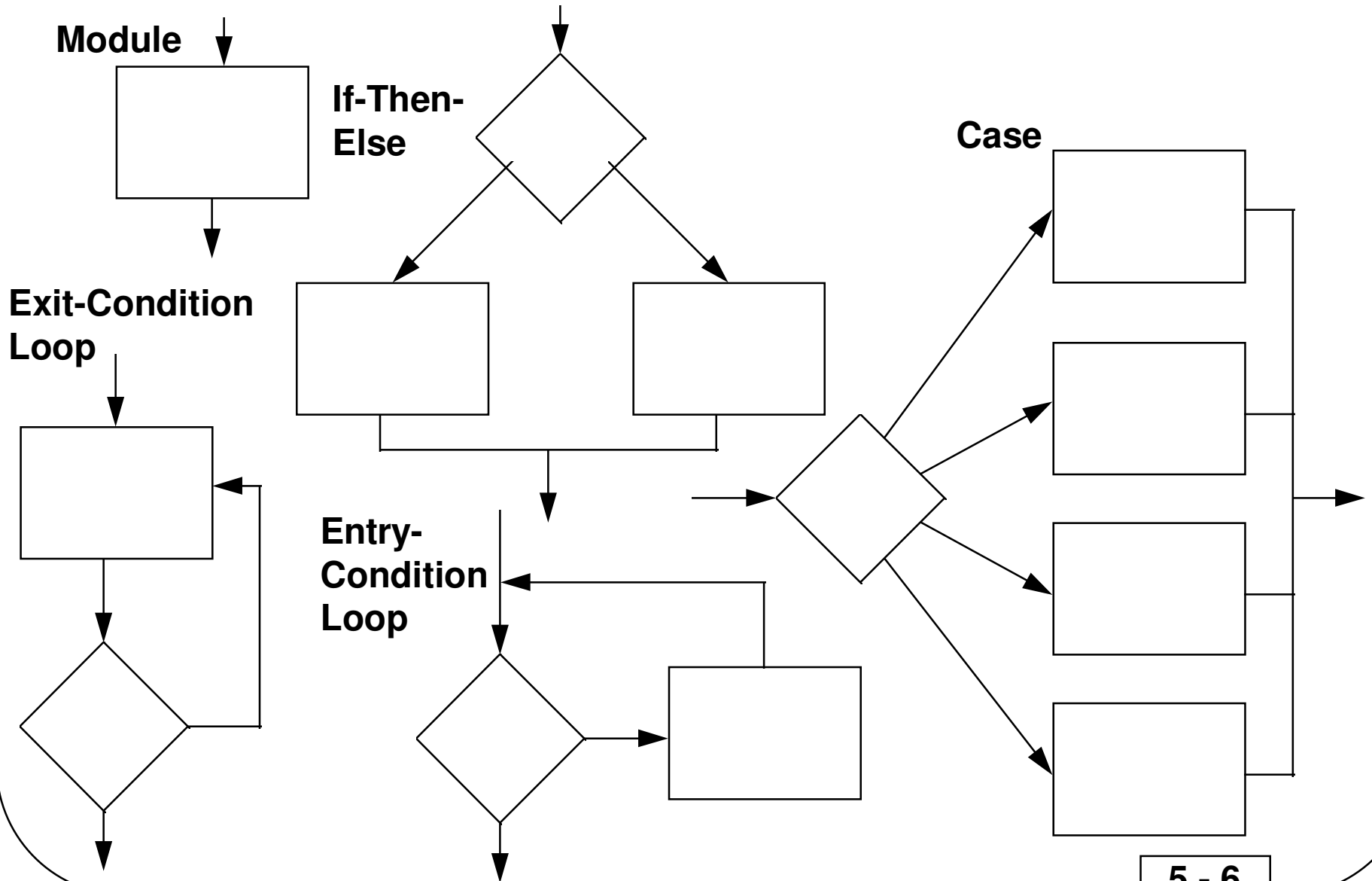
# Goals of Software Engineering



## **Language-Specific Issues**

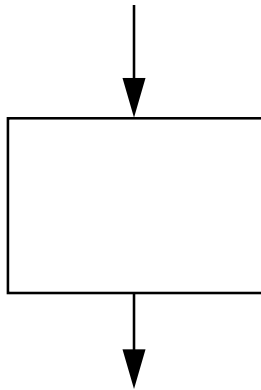
- **Control Structures**
- **Data Typing**
- **Subprograms and Collections**
- **Structured Programming**
- **Object-Oriented Programming**
- **Application Domains**

# Control Structures

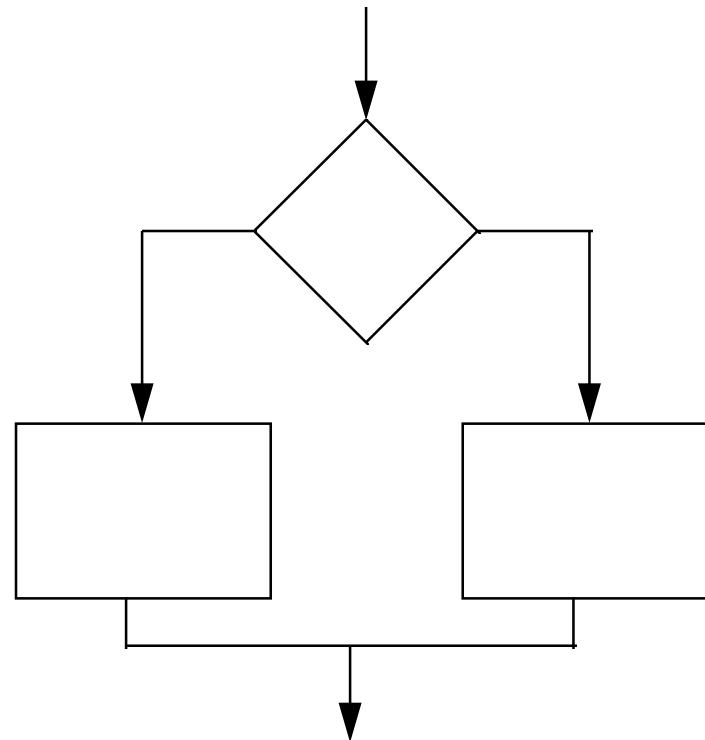


# Control Structures, Continued

Module

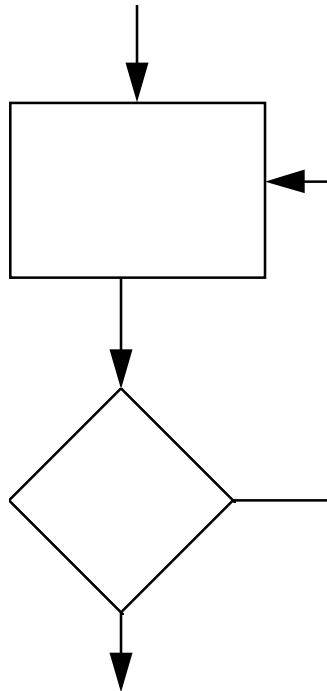


If-Then-Else

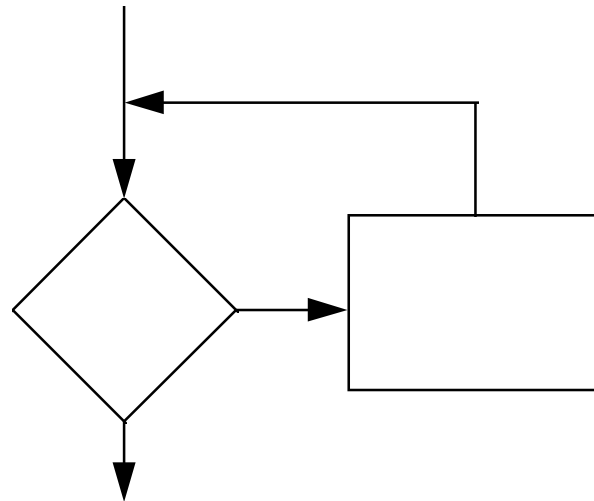


# Control Structures, Continued

Exit-Condition Loop



Entry-Condition Loop

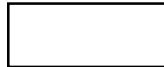




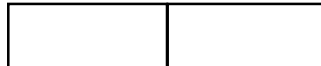
# Data Typing

## Scalar Types

Character/Byte



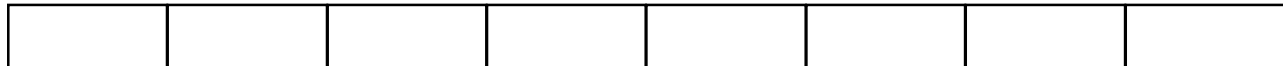
Integer



Float

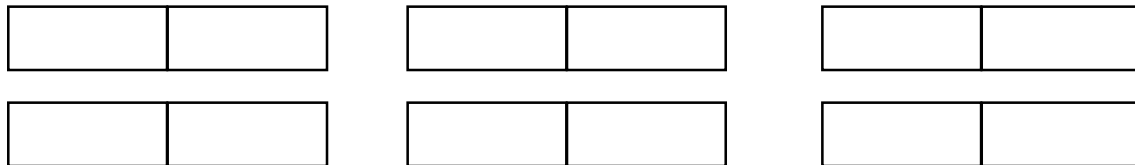


Double Float

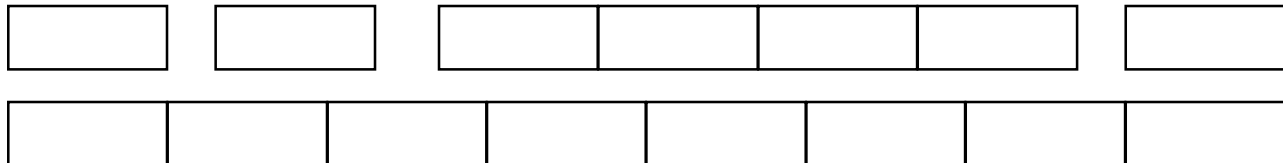


## Aggregate Types

Array



Record



# **Subprograms and Collections**

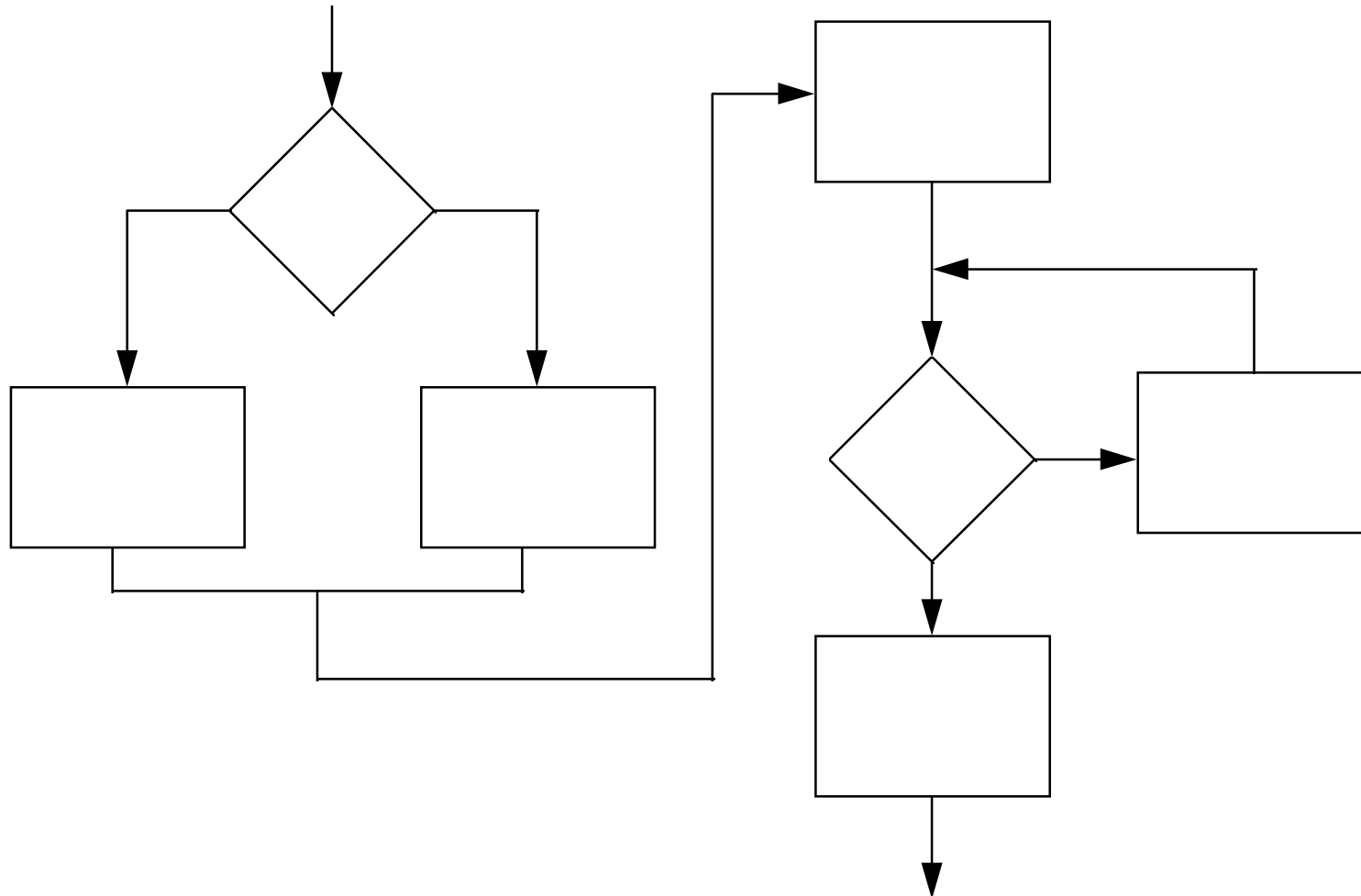
- **Subprograms**

- **Functions** - return a specific value, like the sin of an angle
- **Procedures** - perform a series of operations, returning zero or more values, like reading a line from a file

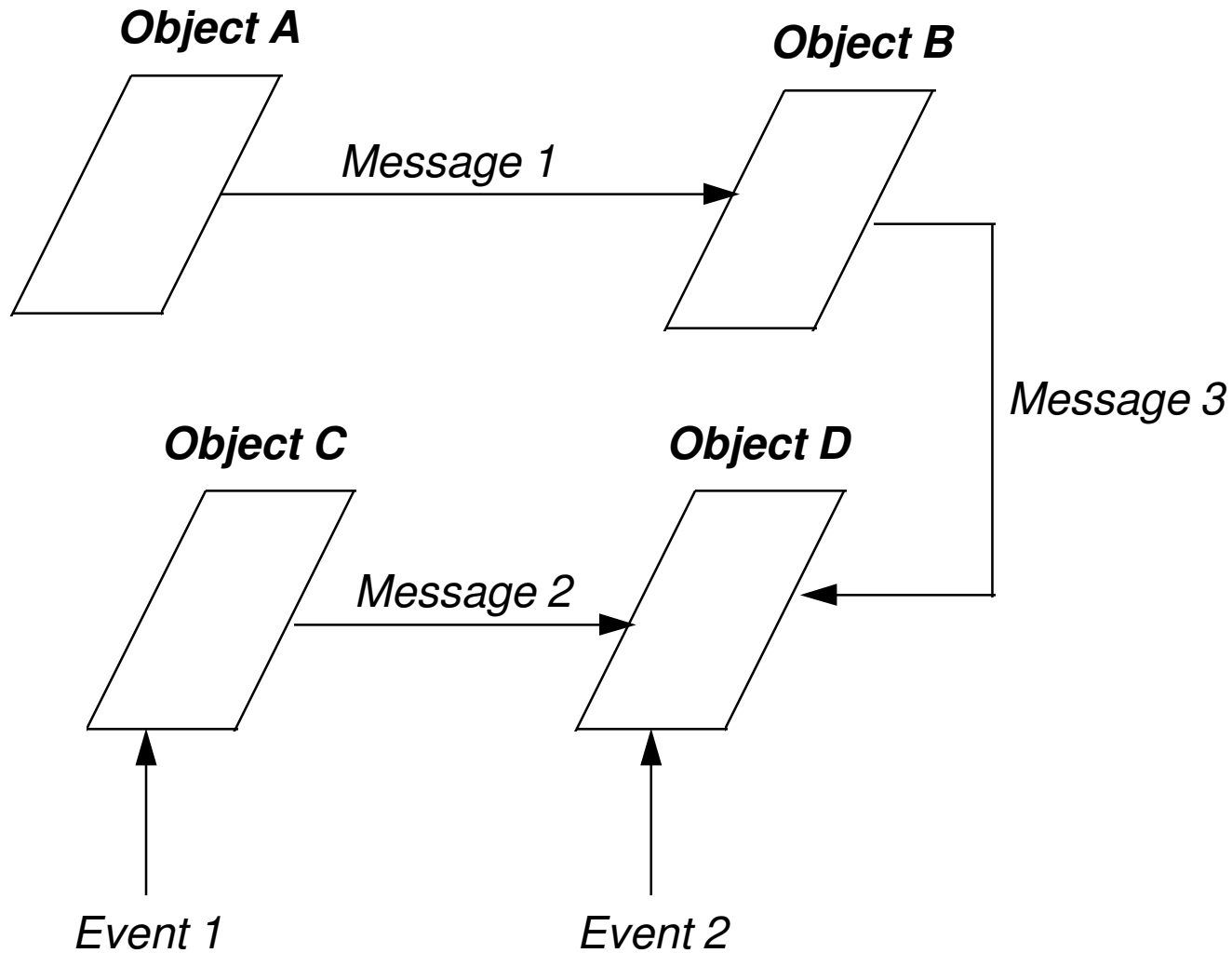
- **Collections**

- **Package** - a group of data, subprograms, and other software constructs
- **Class** - a group of data and subprograms related to a number of similar objects

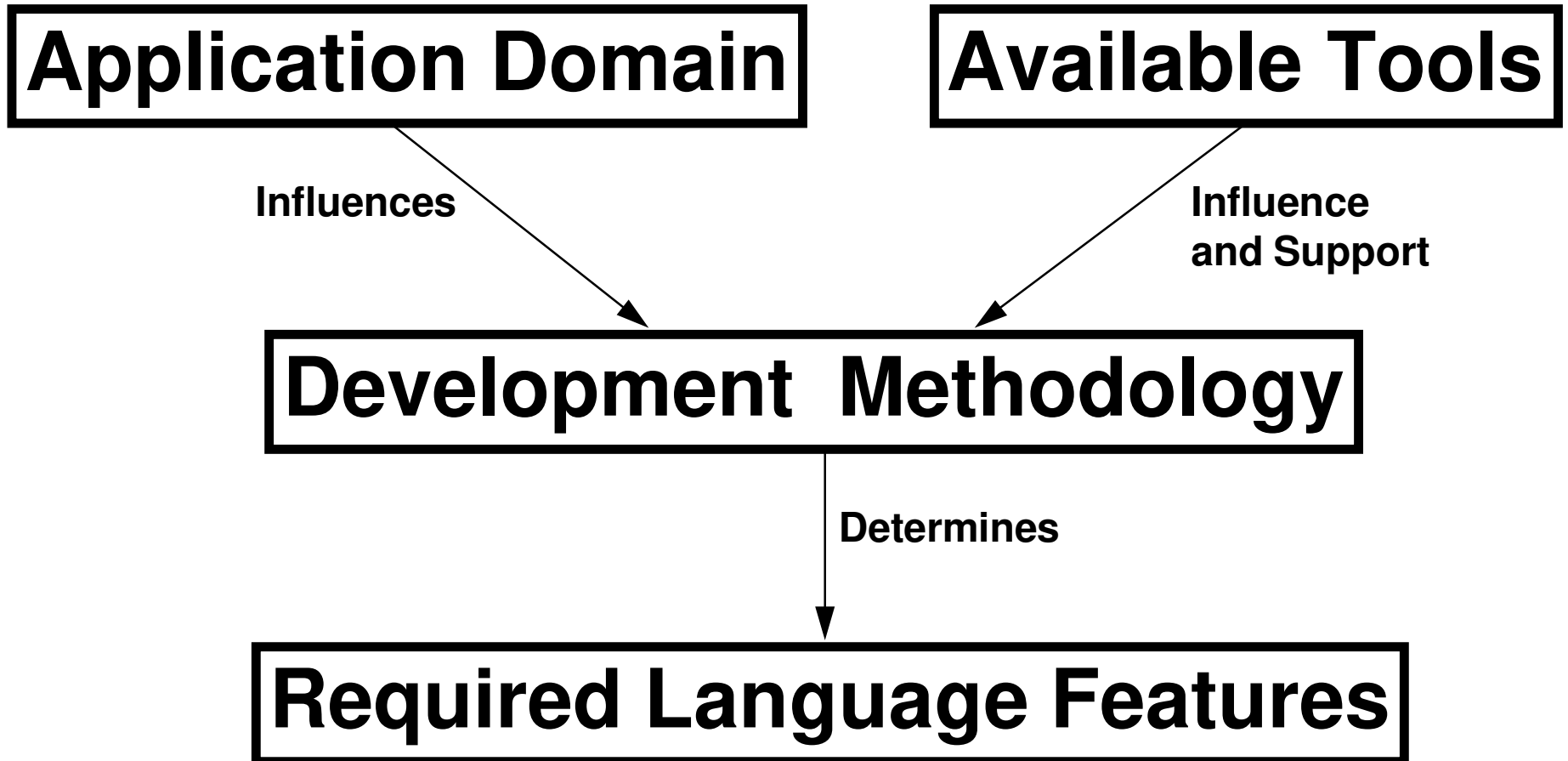
# Structured Programming



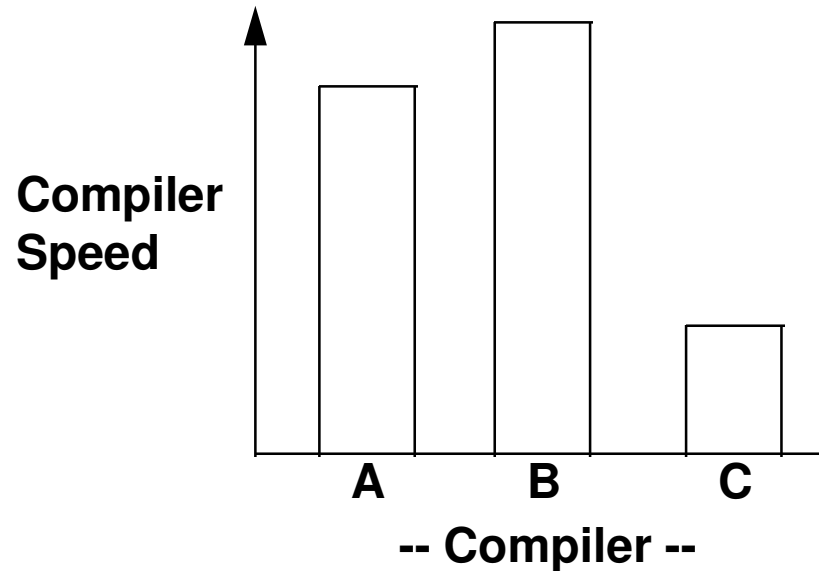
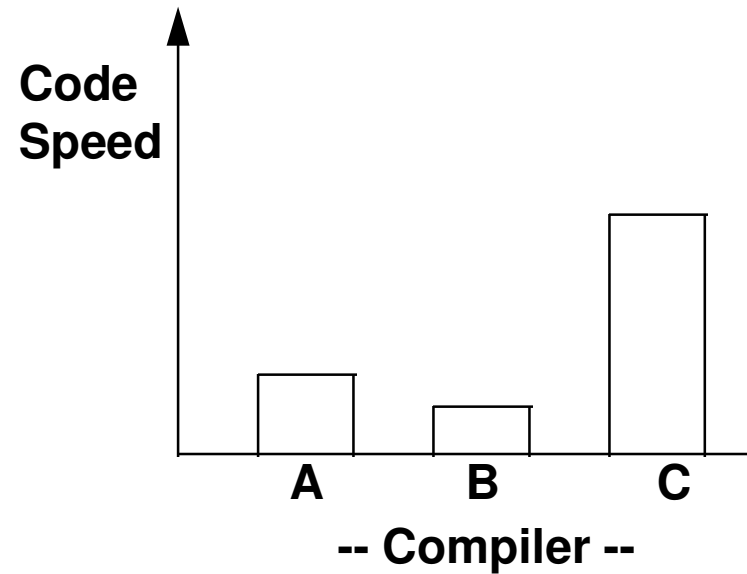
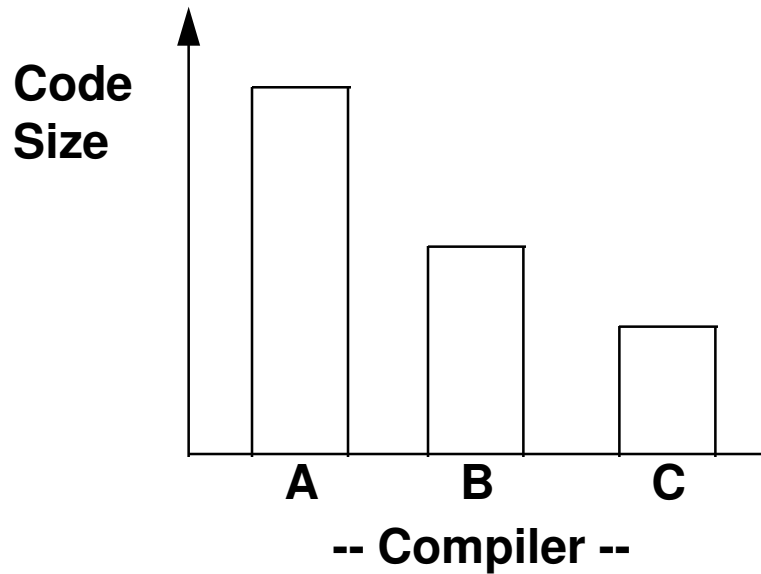
# Object-Oriented Programming



# Application Domains



# Compiler-Specific Issues



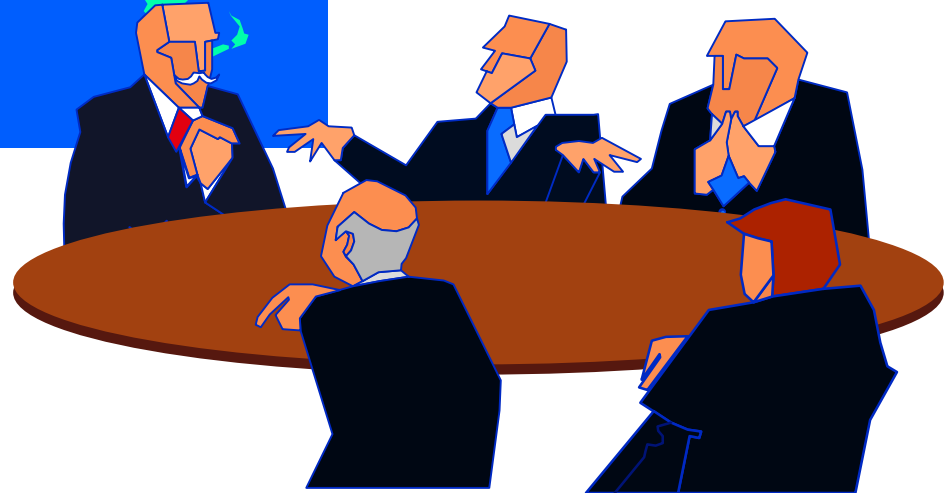
## **Organizational Issues**

- **Culture and Psychological View**
- **Education and Training,  
Resources Required, and Cost**

# Culture and Psychological View



- Culture
- Psychological View
- *Education & Training*
- *Resources Required*
- *Cost*





# Education and Training, Resources Required, and Cost



- *Culture*
- *Psychological View*
- **Education & Training**
- **Resources Required**
- **Cost**



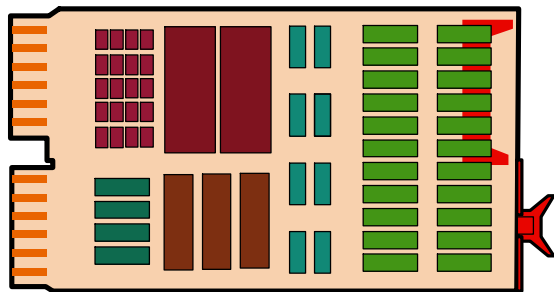
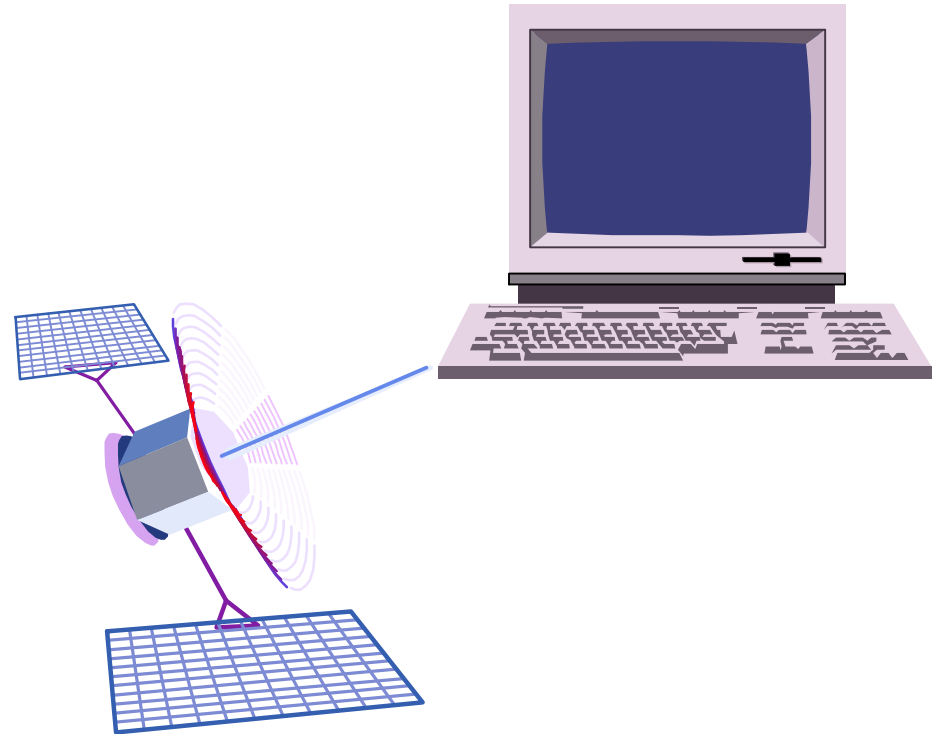
# Language Selection

- **Trends by Application Domain**
- **Criteria for Selection**
- **Assessment**

# Trends by Application Domain

## *Some Application Domains*

- **Systems Software**
- **Real-Time Software**
- **Embedded Software**
- **Business Software**
- **Engineering/Scientific Software**
- **Personal Computer Software**
- **Artificial Intelligence Software**



## *Software Development Across Domains*

- **Structured**
- **Object-Oriented**
- **Fourth Generation**

# **Criteria for Selection**

**Some Criteria --**

- 1. Application domain**
- 2. Algorithmic and computational complexity**
- 3. Environment in which the software will execute**
- 4. Performance considerations**
- 5. Data structure complexity**
- 6. Knowledge of software development staff**
- 7. Availability of a good compiler or cross-compiler**
- 8. Life cycle costs of software development**

# Assessment

## *Assessing a Programming Language - Develop a Yardstick and a Buy-In*

- Determine criteria for selection
- Set weights for each criterion
- Interact with your organization - get a buy-in for the above
- Select an assessment team from various representative groups in your organization
- Perform the assessment analytically
- Brief organization on the results of the assessment and discuss - get a buy-in for the fairness of the assessment
- Reassess if necessary
- Select language and brief the organization

