

# CS201

## Introduction to Programming

### Important mcqs

#### Lec 23 - Pre-processor

1. **What is the pre-processor in C programming?**
- a) A program that executes before the compiler
  - b) A part of the compiler that optimizes code
  - c) A program that executes after the compiler
  - d) A part of the compiler that performs type checking

Answer: a

2. **Which pre-processor directive is used to include header files in a C program?**
- a) #include
  - b) #define
  - c) #ifdef
  - d) #ifndef

Answer: a

3. **What does the #define directive do in C programming?**
- a) Defines a constant value or a macro
  - b) Includes a header file
  - c) Conditionally compiles code
  - d) Expands a pre-defined macro

Answer: a

4. **Which pre-processor directive is used to define a macro in C programming?**
- a) #define
  - b) #include
  - c) #ifdef
  - d) #ifndef

Answer: a

5. **What is the purpose of the #ifdef directive in C programming?**
- a) To include a header file if a macro is defined
  - b) To exclude a block of code if a macro is not defined
  - c) To define a macro
  - d) To conditionally compile code

Answer: b

6. Which pre-processor directive is used to conditionally compile code based on a certain condition?
- a) #define
  - b) #include
  - c) #ifdef
  - d) #if

Answer: d

7. What is the purpose of the #ifndef directive in C programming?
- a) To define a macro
  - b) To include a header file if a macro is not defined
  - c) To exclude a block of code if a macro is defined
  - d) To conditionally compile code

Answer: b

8. Which pre-processor directive is used to undefine a macro in C programming?
- a) #undef
  - b) #define
  - c) #ifdef
  - d) #ifndef

Answer: a

9. What does the #pragma directive do in C programming?
- a) Specifies implementation-specific behavior
  - b) Defines a macro
  - c) Includes a header file
  - d) Expands a pre-defined macro

Answer: a

10. Which pre-processor directive is used to generate code automatically based on templates or other input files?
- a) #define
  - b) #include
  - c) #ifdef
  - d) #pragma

Answer: b

## Lec 24 - Memory Allocation

### 1. What is memory allocation in C programming?

- a) Reserving memory for the program to store data during runtime
- b) Allocating memory for the program during compile time
- c) Storing data in memory during runtime
- d) None of the above

Answer: a) Reserving memory for the program to store data during runtime

### 2. Which of the following is a function used for dynamic memory allocation in C?

- a) calloc()
- b) malloc()
- c) realloc()
- d) All of the above

Answer: d) All of the above

### 3. What is the difference between malloc() and calloc() functions?

- a) malloc() allocates a block of memory of a specified size, while calloc() initializes the memory to 0
- b) calloc() allocates a block of memory of a specified size, while malloc() initializes the memory to 0
- c) malloc() and calloc() are the same function
- d) None of the above

Answer: a) malloc() allocates a block of memory of a specified size, while calloc() initializes the memory to 0

### 4. What happens if malloc() or calloc() is unable to allocate the requested memory?

- a) The program crashes
- b) The function returns NULL
- c) The function returns a negative value
- d) None of the above

Answer: b) The function returns NULL

### 5. What is a memory leak?

- a) When memory is not deallocated after it is no longer needed
- b) When memory is allocated but never used
- c) When memory is allocated and used but not freed after it is no longer needed
- d) None of the above

Answer: a) When memory is not deallocated after it is no longer needed

### 6. Which function is used to free memory allocated by malloc(), calloc(), or realloc()?

- a) dealloc()
- b) free()
- c) remove()
- d) None of the above

Answer: b) free()

7. **What is stack memory allocation?**

- a) Reserving memory for the program during runtime
- b) Allocating memory for the program during compile time
- c) Storing data in memory during runtime
- d) None of the above

Answer: b) Allocating memory for the program during compile time

8. **What is heap memory allocation?**

- a) Reserving memory for the program during runtime
- b) Allocating memory for the program during compile time
- c) Storing data in memory during runtime
- d) None of the above

Answer: a) Reserving memory for the program during runtime

9. **What is the purpose of the realloc() function in C programming?**

- a) To allocate a new block of memory
- b) To deallocate a block of memory
- c) To resize an existing block of memory
- d) None of the above

Answer: c) To resize an existing block of memory

10. **What is the potential risk of not properly managing memory allocation in C programming?**

- a) Memory leaks
- b) Memory fragmentation
- c) Program crashes
- d) All of the above

Answer: d) All of the above

## Lec 25 - Lecture Overview

1. **What is a lecture overview?**
- A. A detailed analysis of the lecture
  - B. A brief summary of the lecture
  - C. A list of questions related to the lecture
  - D. A video recording of the lecture

**Answer: B. A brief summary of the lecture**

2. **What is the purpose of a lecture overview?**
- A. To summarize the lecture content
  - B. To provide a list of exam questions
  - C. To introduce the lecturer
  - D. To provide a transcript of the lecture

**Answer: A. To summarize the lecture content**

3. **Which of the following is NOT typically included in a lecture overview?**
- A. Key concepts
  - B. Relevant background information
  - C. Examples and case studies
  - D. In-depth analysis of the lecture

**Answer: D. In-depth analysis of the lecture**

4. **What is the benefit of a good lecture overview for students?**
- A. Helps them prepare for the lecture
  - B. Allows them to skip the lecture
  - C. Provides them with exam answers
  - D. None of the above

**Answer: A. Helps them prepare for the lecture**

5. **Who typically provides the lecture overview?**
- A. The lecturer
  - B. A teaching assistant
  - C. A student
  - D. An outside expert

**Answer: A. The lecturer**

6. **Which of the following is an example of relevant background information that might be included in a lecture overview?**
- A. The lecturer's personal life
  - B. A historical context for the topic
  - C. Student grades on previous assignments
  - D. A list of extracurricular activities

**Answer: B. A historical context for the topic**

7. **How can a lecture overview help students stay focused during the lecture?**

- A. By providing a list of questions
- B. By highlighting key concepts
- C. By providing a transcript of the lecture
- D. By providing a quiz at the end of the lecture

**Answer: B. By highlighting key concepts**

8. **Which of the following is NOT a common format for a lecture overview?**

- A. A written summary
- B. A PowerPoint presentation
- C. A video recording
- D. A list of exam questions

**Answer: D. A list of exam questions**

9. **How can a lecture overview benefit the lecturer?**

- A. By providing feedback on their teaching style
- B. By saving time during the lecture
- C. By improving student engagement
- D. All of the above

**Answer: D. All of the above**

10. **What is the primary goal of a lecture overview?**

- A. To entertain the audience
- B. To provide a detailed analysis of the topic
- C. To summarize the lecture content
- D. To introduce the lecturer

**Answer: C. To summarize the lecture content**

## Lec 26 - Classes and Objects

1. **What is a class in object-oriented programming?**

- a. An instance of an object
- b. A function that returns a value
- c. A blueprint or template for creating objects
- d. A data type used for arithmetic operations

**Answer: c. A blueprint or template for creating objects**

2. **Which of the following is NOT an advantage of using classes and objects?**

- a. Reusability of code
- b. Encapsulation of data and behavior
- c. Easier to understand and maintain code
- d. Slower program execution

**Answer: d. Slower program execution**

3. **Which keyword is used to create an object of a class in Python?**

- a. create
- b. new
- c. make
- d. None of the above

**Answer: d. None of the above (In Python, objects are created simply by calling the class as if it were a function)**

4. **Which of the following is a characteristic of an object?**

- a. Data
- b. Behavior
- c. Both A and B
- d. None of the above

**Answer: c. Both A and B**

5. **Which method is called when an object is created?**

- a. **init**
- b. new
- c. create
- d. None of the above

**Answer: b. new**

6. **Which of the following is a feature of object-oriented programming?**

- a. Inheritance
- b. Encapsulation
- c. Polymorphism
- d. All of the above

**Answer: d. All of the above**

7. **What is inheritance in object-oriented programming?**

- a. The process of creating a new object from an existing object
- b. The ability to create a new class from an existing class
- c. The process of adding new methods to a class
- d. The ability to call methods from another class

**Answer: b. The ability to create a new class from an existing class**

8. **What is encapsulation in object-oriented programming?**

- a. The ability to hide the internal workings of an object
- b. The process of creating a new object from an existing object
- c. The ability to call methods from another class
- d. The process of adding new methods to a class

**Answer: a. The ability to hide the internal workings of an object**

9. **What is polymorphism in object-oriented programming?**

- a. The ability to create a new class from an existing class
- b. The process of adding new methods to a class
- c. The ability of objects of different classes to be treated as if they were of the same class
- d. The ability to call methods from another class

**Answer: c. The ability of objects of different classes to be treated as if they were of the same class**

10. **What is the difference between a class and an object?**

- a. A class is a blueprint for creating objects, while an object is an instance of a class.
- b. A class is an instance of an object, while an object is a blueprint for creating classes.
- c. A class and an object are the same thing.
- d. None of the above.

**Answer: a. A class is a blueprint for creating objects, while an object is an instance of a class.**

## Lec 27 - Classes And Objects

### 1. What is a class in object-oriented programming?

- a) An instance of an object
- b) A blueprint or template for creating objects
- c) A method for manipulating data
- d) A data structure for storing information

Answer: b

### 2. What is an object in object-oriented programming?

- a) A blueprint or template for creating classes
- b) A method for manipulating data
- c) An instance of a class
- d) A data structure for storing information

Answer: c

### 3. What is encapsulation in object-oriented programming?

- a) The ability of objects of different classes to be treated as if they were of the same class
- b) The ability to hide the internal workings of an object from the outside world
- c) The ability to create a new class from an existing class
- d) The ability to initialize the attributes of an object

Answer: b

### 4. What is inheritance in object-oriented programming?

- a) The ability of objects of different classes to be treated as if they were of the same class
- b) The ability to hide the internal workings of an object from the outside world
- c) The ability to create a new class from an existing class
- d) The ability to initialize the attributes of an object

Answer: c

### 5. What is polymorphism in object-oriented programming?

- a) The ability of objects of different classes to be treated as if they were of the same class
- b) The ability to hide the internal workings of an object from the outside world
- c) The ability to create a new class from an existing class
- d) The ability to initialize the attributes of an object

Answer: a

### 6. What is a constructor in object-oriented programming?

- a) A method for manipulating data
- b) An instance of a class
- c) A data structure for storing information
- d) A method used to initialize objects

Answer: d

7. **What is the difference between a class and an object?**

- a) A class is a blueprint or template, while an object is an instance of that class
- b) A class is an instance of an object, while an object is a blueprint or template
- c) A class and an object are the same thing
- d) A class is a method, while an object is a data structure

**Answer: a**

8. **What is an attribute in object-oriented programming?**

- a) A method for manipulating data
- b) An instance of a class
- c) A data variable that describes the state of an object
- d) A data structure for storing information

**Answer: c**

9. **What is a method in object-oriented programming?**

- a) A data variable that describes the state of an object
- b) An instance of a class
- c) A function that defines the behavior of an object
- d) A data structure for storing information

**Answer: c**

10. **What is the advantage of using classes and objects in object-oriented programming?**

- a) Encapsulation of data and functionality
- b) Inheritance and polymorphism for code reuse
- c) A more flexible and dynamic code structure
- d) All of the above

**Answer: d**

## Lec 28 - Lecture Overview

1. **What is the purpose of a lecture overview?**
- A. To provide a detailed summary of the lecture material
  - B. To help students prepare for upcoming assignments or exams
  - C. To replace the need for taking notes during the lecture
  - D. None of the above

Answer: B

2. **Which of the following is a benefit of using a lecture overview?**
- A. Helps students to take better notes during the lecture
  - B. Provides a reference for future study and review
  - C. Allows students to skip attending the lecture
  - D. Both A and B

Answer: D

3. **Who benefits from a lecture overview?**
- A. Students
  - B. Instructors
  - C. Both A and B
  - D. None of the above

Answer: A

4. **What information is typically included in a lecture overview?**
- A. Key points and concepts covered in the lecture
  - B. Objectives of the lecture
  - C. Examples and illustrations used in the lecture
  - D. All of the above

Answer: D

5. **When should a student refer to a lecture overview?**
- A. During the lecture
  - B. After the lecture
  - C. Both A and B
  - D. None of the above

Answer: C

6. **Which of the following is a disadvantage of using a lecture overview?**
- A. It can be time-consuming to create
  - B. It can replace the need for attending the lecture
  - C. It can be difficult to understand without attending the lecture
  - D. None of the above

Answer: A

7. **How can a lecture overview help students stay focused during the lecture?**

- A. By providing a preview of the key topics and concepts
- B. By allowing students to skip parts of the lecture
- C. By replacing the need for taking notes during the lecture
- D. None of the above

**Answer: A**

8. **What is the primary goal of a lecture overview?**

- A. To replace the need for attending the lecture
- B. To provide a detailed summary of the lecture material
- C. To help students prepare for upcoming assignments or exams
- D. None of the above

**Answer: C**

9. **Which of the following is an example of a lecture overview?**

- A. A detailed transcript of the lecture
- B. A brief summary of the key points covered in the lecture
- C. A list of all the examples and illustrations used in the lecture
- D. None of the above

**Answer: B**

10. **How can a lecture overview benefit an instructor?**

- A. By allowing the instructor to skip parts of the lecture
- B. By providing a reference for future lecture planning
- C. By replacing the need for the instructor to teach the material
- D. None of the above

**Answer: B**

## Lec 29 - Friend functions

1. **What is a friend function in C++?**

- A) A function declared inside a class that can only access public members
- B) A member function that can access private and protected members of another class
- C) A non-member function that is given access to the private and protected members of a class
- D) A global function that is restricted from accessing the data members of a class

**Answer: C**

2. **Which keyword is used to declare a friend function inside a class?**

- A) friend
- B) private
- C) public
- D) protected

**Answer: A**

3. **Can a friend function access the private members of a class?**

- A) Yes
- B) No

**Answer: A**

4. **Which of the following is an advantage of using friend functions?**

- A) It simplifies the implementation of certain operations
- B) It enhances encapsulation of class data
- C) It improves program maintainability
- D) It reduces code complexity

**Answer: A**

5. **Which of the following is a disadvantage of using friend functions?**

- A) It reduces encapsulation and can make the program harder to maintain
- B) It requires the use of complex syntax to declare and define the function
- C) It can lead to security vulnerabilities in the program
- D) It can only be used for simple operations and cannot handle complex tasks

**Answer: A**

6. **Can a friend function be a member function of another class?**

- A) Yes
- B) No

**Answer: A**

7. **What is the access level of a friend function in relation to the class it is declared in?**

- A) Private
- B) Protected
- C) Public

D) It is not a member of the class, so access level does not apply

**Answer: D**

8. **Can a friend function be overloaded?**

- A) Yes
- B) No

**Answer: A**

9. **Can a friend function be declared in the private section of a class?**

- A) Yes
- B) No

**Answer: A**

10. **Which of the following is an example of a scenario where a friend function would be useful?**

- A) To add two numbers together
- B) To sort a list of objects
- C) To calculate the distance between two objects in a class
- D) To print the contents of a private data member

**Answer: D**

## Lec 30 - Reference data type

### 1. What is a reference data type in C++?

- a) It is a variable that stores the memory address of another variable
- b) It is a variable that refers to another variable by name
- c) It is a variable that can be assigned a null value
- d) It is a variable that cannot be passed to a function

Answer: b

### 2. What is the difference between a pointer and a reference in C++?

- a) Pointers can be reassigned to point to different variables, while references cannot.
- b) References can be null, while pointers cannot.
- c) Pointers are used to pass variables by reference, while references are used to pass variables by value.
- d) There is no difference between a pointer and a reference in C++.

Answer: a

### 3. Can a reference be declared without being initialized?

- a) Yes, a reference can be declared without being initialized.
- b) No, a reference must be initialized when it is declared.
- c) It depends on the data type of the reference.
- d) It depends on the scope of the reference.

Answer: b

### 4. What is the benefit of passing parameters by reference in a function?

- a) It saves memory by not creating a copy of the variable.
- b) It allows the function to modify the original variable.
- c) It makes the code more readable.
- d) It makes the code faster.

Answer: b

### 5. What happens if a reference is assigned to a new variable?

- a) The original variable is deleted.
- b) The new variable becomes an alias for the original variable.
- c) A new copy of the original variable is created.
- d) The program crashes.

Answer: b

### 6. Can a reference be used as a return type for a function?

- a) Yes, a reference can be used as a return type for a function.
- b) No, a reference cannot be used as a return type for a function.
- c) It depends on the data type of the reference.
- d) It depends on the scope of the reference.

Answer: a

7. **What is the syntax for declaring a reference variable in C++?**

- a) `int& x;`
- b) `int* x;`
- c) `int x&;`
- d) `int& x = y;`

**Answer: d**

8. **Can a reference refer to a const variable?**

- a) Yes, a reference can refer to a const variable.
- b) No, a reference cannot refer to a const variable.
- c) It depends on the data type of the reference.
- d) It depends on the scope of the reference.

**Answer: a**

9. **What is the difference between a const reference and a non-const reference?**

- a) A const reference cannot be modified, while a non-const reference can.
- b) A non-const reference cannot be modified, while a const reference can.
- c) There is no difference between a const reference and a non-const reference.
- d) A const reference cannot refer to a non-const variable.

**Answer: a**

10. **Can a reference refer to a temporary object?**

- a) Yes, a reference can refer to a temporary object.
- b) No, a reference cannot refer to a temporary object.
- c) It depends on the data type of the reference.
- d) It depends on the scope of the reference.

**Answer: a**

## Lec 31 - Lecture Overview

1. **Which of the following is a characteristic of a good lecture?**

- a) It is scripted and read word-for-word.
- b) It is monotone and lacks enthusiasm.
- c) It is interactive and engages the audience.
- d) It is rushed and covers too much material in a short time.

**Answer: c) It is interactive and engages the audience.**

2. **Which of the following is a benefit of using visual aids during a lecture?**

- a) They can distract the audience from the main message.
- b) They can help to reinforce key points and increase retention.
- c) They are not necessary for an effective lecture.
- d) They can be replaced by written handouts.

**Answer: b) They can help to reinforce key points and increase retention.**

3. **What is the recommended amount of time that a lecture should last?**

- a) 30 minutes
- b) 60 minutes
- c) 90 minutes
- d) 120 minutes

**Answer: c) 90 minutes**

4. **Which of the following is a strategy for dealing with challenging questions during a lecture?**

- a) Ignoring the question and moving on to the next topic.
- b) Admitting that you don't know the answer.
- c) Repeating the question back to the audience to ensure clarity.
- d) Providing a lengthy and complex answer.

**Answer: c) Repeating the question back to the audience to ensure clarity.**

5. **What is the primary purpose of a lecture?**

- a) To provide information to the audience.
- b) To entertain the audience.
- c) To test the audience's knowledge.
- d) To sell a product or service.

**Answer: a) To provide information to the audience.**

6. **What is an effective way to begin a lecture?**

- a) By telling a joke or personal story.
- b) By immediately diving into the content.
- c) By asking the audience a question to get them thinking.
- d) By introducing yourself and your credentials.

**Answer: c) By asking the audience a question to get them thinking.**

7. **What is a common mistake that some lecturers make?**

- a) Focusing too much on one topic and neglecting others.
- b) Speaking too quickly and not allowing time for questions.
- c) Using too many visual aids and overwhelming the audience.
- d) Ignoring the audience and speaking only to themselves.

**Answer: b) Speaking too quickly and not allowing time for questions.**

8. **Which of the following is an effective way to keep the audience engaged during a lecture?**

- a) Speaking in a monotone voice.
- b) Avoiding eye contact with the audience.
- c) Asking questions and encouraging participation.
- d) Reading from a script without deviation.

**Answer: c) Asking questions and encouraging participation.**

9. **What is an effective way to conclude a lecture?**

- a) Abruptly stopping and exiting the stage.
- b) Summarizing key points and leaving time for questions.
- c) Making a sales pitch for a product or service.
- d) Ignoring questions and moving on to the next topic.

**Answer: b) Summarizing key points and leaving time for questions.**

10. **Which of the following is a way to improve as a lecturer?**

- a) Avoiding any negative feedback or criticism.
- b) Refusing to adjust your approach to fit the audience.
- c) Continuously seeking feedback and making adjustments as needed.
- d) Using the same lecture format and content for every audience.

**Answer: c) Continuously seeking feedback and making adjustments as needed.**

## Lec 32 - Recap

### 1. What is Recap?

- a) A new social media platform
- b) A type of dance
- c) A summary or review of events, information, or experiences
- d) A type of food

**Solution: c) A summary or review of events, information, or experiences**

### 2. What are the benefits of Recap?

- a) Reinforces learning
- b) Tracks progress
- c) Identifies areas for improvement
- d) All of the above

**Solution: d) All of the above**

### 3. In which contexts can Recap be beneficial?

- a) Academic
- b) Professional
- c) Personal
- d) All of the above

**Solution: d) All of the above**

### 4. What is the purpose of a Recap?

- a) To confuse the reader
- b) To summarize or review events, information, or experiences
- c) To add unnecessary information
- d) To make the information more difficult to understand

**Solution: b) To summarize or review events, information, or experiences**

### 5. Can a Recap be shared with others?

- a) Yes
- b) No

**Solution: a) Yes**

### 6. How can a Recap help with communication?

- a) By making information more difficult to understand
- b) By reinforcing learning
- c) By summarizing and consolidating key points
- d) By adding unnecessary information

**Solution: c) By summarizing and consolidating key points**

### 7. What is the main goal of a Recap?

- a) To confuse the reader

- b) To make the information more difficult to understand
- c) To reinforce learning and consolidate key points
- d) To make the information less accessible

**Solution: c) To reinforce learning and consolidate key points**

**8. What is the difference between Recap and Summary?**

- a) There is no difference
- b) Recap is shorter than Summary
- c) Summary is more detailed than Recap
- d) Recap focuses on key points while Summary provides a comprehensive overview

**Solution: d) Recap focuses on key points while Summary provides a comprehensive overview**

**9. How can Recaps be used in professional contexts?**

- a) To track progress and identify areas for improvement
- b) To share with colleagues for collaboration and feedback
- c) To reinforce learning
- d) All of the above

**Solution: d) All of the above**

**10. How can Recaps be used in personal contexts?**

- a) To reinforce learning
- b) To track progress
- c) To identify areas for improvement
- d) All of the above

**Solution: d) All of the above**

## Lec 33 - Operator Overloading

1. **What is operator overloading?**

- A. The process of creating new operators
- B. The process of redefining existing operators
- C. The process of removing existing operators
- D. The process of renaming existing operators

**Answer: B. The process of redefining existing operators**

2. **Which of the following operators cannot be overloaded in C++?**

- A. +
- B. -
- C. \*
- D. ::

**Answer: D. ::**

3. **Which of the following is not a unary operator?**

- A. +
- B. -
- C. \*
- D. /

**Answer: D. /**

4. **Which of the following operators must be overloaded as a member function?**

- A. =
- B. []
- C. ()
- D. <<

**Answer: C. ()**

5. **Which of the following operators must be overloaded as a friend function?**

- A. =
- B. +
- C. ++
- D. []

**Answer: D. []**

6. **Which of the following is the correct syntax for overloading the addition operator in C++?**

- A. operator add()
- B. operator +()
- C. operator +(int)
- D. operator add(int)

**Answer: B. operator +()**

7. Which of the following operators is used to access the elements of an array?

- A. []
- B. ()
- C. =
- D. \*

Answer: A. []

8. Which of the following operators is used to define the behavior of a user-defined object when it is converted to a basic data type?

- A. ()
- B. =
- C. <<
- D. >>

Answer: A. ()

9. Which of the following operators cannot be overloaded as a friend function?

- A. +
- B. -
- C. =
- D. ()

Answer: C. =

10. Which of the following is not a binary operator?

- A. +
- B. \*
- C. ++
- D. /

Answer: C. ++

## Lec 34 - Arrays of Objects

1. **What is an array of objects in programming?**
- A. An array that stores integers
  - B. An array that stores characters
  - C. An array that stores multiple objects of the same class
  - D. An array that stores multiple functions

**Answer: C**

2. **How do you declare an array of objects in C++?**
- A. `int arr[];`
  - B. `char arr[];`
  - C. `ClassName arr[];`
  - D. `Object arr[];`

**Answer: C**

3. **How do you initialize an array of objects in C++?**
- A. Using a loop
  - B. Using a function
  - C. Using a constructor
  - D. Using an if statement

**Answer: C**

4. **Can you access individual elements of an array of objects using array index notation?**
- A. Yes
  - B. No

**Answer: A**

5. **How do you access a member function of an object in an array of objects?**
- A. `objectName.memberFunction()`
  - B. `arrayName[index].memberFunction()`
  - C. `objectName[index].memberFunction()`
  - D. `arrayName.memberFunction()`

**Answer: B**

6. **How do you access a data member of an object in an array of objects?**
- A. `objectName.dataMember`
  - B. `arrayName[index].dataMember`
  - C. `objectName[index].dataMember`
  - D. `arrayName.dataMember`

**Answer: B**

7. **What is the size of an array of objects?**
- A. It depends on the type of objects stored in the array

- B. It depends on the number of elements in the array
- C. It depends on the memory allocated for the array
- D. It depends on the size of the objects stored in the array

**Answer: B**

8. **Can you sort an array of objects in C++ using the sort() function?**

- A. Yes
- B. No

**Answer: A**

9. **How do you delete an object from an array of objects?**

- A. Using the delete operator
- B. Using the remove() function
- C. Using the pop() function
- D. It is not possible to delete an object from an array of objects

**Answer: D**

10. **What is the benefit of using an array of objects in programming?**

- A. It allows for the manipulation of multiple objects at once
- B. It reduces the size of the program
- C. It increases the speed of the program
- D. It makes the code more complex

**Answer: A**

## Lec 35 - Streams

1. Which header file is used for handling input/output operations in C++?

- A) iostream
- B) string
- C) vector
- D) algorithm

Answer: A

2. Which of the following is not a standard stream in C++?

- A) cin
- B) cout
- C) cerr
- D) clog

Answer: D

3. Which operator is used for insertion (output) operation with streams?

- A) <<
- B) >>
- C) ||
- D) &&

Answer: A

4. Which operator is used for extraction (input) operation with streams?

- A) <<
- B) >>
- C) ||
- D) &&

Answer: B

5. What is the default mode for file stream in C++?

- A) read
- B) write
- C) append
- D) binary

Answer: A

6. Which stream is used for handling errors during input/output operations?

- A) cin
- B) cout
- C) cerr
- D) clog

Answer: C

7. Which function is used to open a file for reading in C++?

- A) open()
- B) read()
- C) write()
- D) close()

Answer: A

8. Which function is used to close a file in C++?

- A) open()
- B) read()
- C) write()
- D) close()

Answer: D

9. Which function is used to check if a file is open or not in C++?

- A) is\_open()
- B) read()
- C) write()
- D) close()

Answer: A

10. Which function is used to read a line of text from a file in C++?

- A) getline()
- B) read()
- C) write()
- D) close()

Answer: A

## Lec 36 - Stream Manipulations

1. Which of the following stream manipulators is used to set the width of output data?
- A. setw()
  - B. setwidth()
  - C. setfill()
  - D. setprecision()

Answer: A. setw()

2. What is the purpose of the setprecision() stream manipulator?
- A. To set the width of output data
  - B. To set the number of decimal places for output data
  - C. To skip whitespace when reading input data
  - D. To set the fill character for output data

Answer: B. To set the number of decimal places for output data

3. Which of the following stream manipulators is used to control the format of numeric output?
- A. setprecision()
  - B. setformat()
  - C. setfill()
  - D. setiosflags()

Answer: D. setiosflags()

4. What is the purpose of the showpoint stream flag?
- A. To show the decimal point for floating-point output
  - B. To show the sign of negative numbers
  - C. To skip leading whitespace when reading input data
  - D. To show trailing zeros for floating-point output

Answer: A. To show the decimal point for floating-point output

5. Which of the following stream manipulators is used to skip whitespace when reading input data?
- A. setw()
  - B. setwidth()
  - C. skipws
  - D. noskipws

Answer: C. skipws

6. What is the purpose of the setfill() stream manipulator?
- A. To set the width of output data
  - B. To set the number of decimal places for output data
  - C. To skip whitespace when reading input data
  - D. To set the fill character for output data

Answer: D. To set the fill character for output data

7. Which of the following stream manipulators is used to reset the format flags for a stream?
- A. resetiosflags()
  - B. setiosflags()
  - C. clear()
  - D. setbase()

Answer: A. resetiosflags()

8. What is the purpose of the setw() and setfill() manipulators used together?
- A. To set the width of output data
  - B. To set the number of decimal places for output data
  - C. To skip whitespace when reading input data
  - D. To set the fill character and width for output data

Answer: D. To set the fill character and width for output data

9. Which of the following stream manipulators is used to control the format of output data?
- A. setprecision()
  - B. setiosflags()
  - C. setfill()
  - D. setw()

Answer: B. setiosflags()

10. What is the purpose of the noshowpoint stream flag?
- A. To hide the decimal point for floating-point output
  - B. To hide the sign of negative numbers
  - C. To skip trailing whitespace when reading input data
  - D. To hide trailing zeros for floating-point output

Answer: A. To hide the decimal point for floating-point output

## Lec 37 - Overloading Insertion and Extraction Operators

1. Which operator is used for input in C++?

- a) <<
- b) >>
- c) &
- d) \*

Answer: b) >>

2. Which operator is used for output in C++?

- a) +
- b) \*
- c) >>
- d) <<

Answer: d) <<

3. Which function is used to overload the extraction operator?

- a) operator-
- b) operator>>
- c) operator<
- d) operator[]

Answer: b) operator>>

4. Which function is used to overload the insertion operator?

- a) operator<
- b) operator>>
- c) operator-
- d) operator<<

Answer: d) operator<<

5. What is the return type of the overloaded insertion operator?

- a) void
- b) int
- c) ostream&
- d) istream&

Answer: c) ostream&

6. What is the return type of the overloaded extraction operator?

- a) void
- b) int
- c) ostream&
- d) istream&

Answer: d) istream&

7. **What is the first parameter of the overloaded insertion operator?**

- a) ostream&
- b) istream&
- c) int
- d) char

**Answer: a) ostream&**

8. **What is the first parameter of the overloaded extraction operator?**

- a) ostream&
- b) istream&
- c) int
- d) char

**Answer: b) istream&**

9. **What is the second parameter of the overloaded insertion operator?**

- a) ostream&
- b) istream&
- c) int
- d) const Object&

**Answer: d) const Object&**

10. **What is the second parameter of the overloaded extraction operator?**

- a) ostream&
- b) istream&
- c) int
- d) Object&

**Answer: d) Object&**

## Lec 38 - User Defined Manipulator

1. **What is a user-defined manipulator in C++?**

- A. A function that modifies the input of a stream
- B. A function that modifies the output of a stream
- C. A function that sorts an array of integers
- D. A function that calculates the factorial of a number

Answer: B

2. **How are user-defined manipulators defined in C++?**

- A. As member functions of a class
- B. As global functions outside of any class
- C. As friend functions of a class
- D. As virtual functions of a class

Answer: B

3. **What is the purpose of a user-defined manipulator?**

- A. To read input from a stream
- B. To modify the output of a stream
- C. To create a new stream object
- D. To delete data from a stream

Answer: B

4. **Which operator is used to call a user-defined manipulator?**

- A. >>
- B. <<
- C. +
- D. -

Answer: B

5. **What is the syntax for defining a user-defined manipulator function?**

- A. `void manipulator(ostream& stream);`
- B. `ostream& operator<<(ostream& stream, manipulator fn);`
- C. `int manipulator(int x);`
- D. `void operator<<(ostream& stream, manipulator fn);`

Answer: B

6. **What is the return type of a user-defined manipulator function?**

- A. void
- B. int
- C. char
- D. ostream&

Answer: D

7. **What is the purpose of the `std::setw()` function?**

- A. To set the width of the output field
- B. To set the precision of the output field
- C. To set the fill character of the output field
- D. To set the format flags of the output field

**Answer: A**

8. **What is the purpose of the `std::setfill()` function?**

- A. To set the width of the output field
- B. To set the precision of the output field
- C. To set the fill character of the output field
- D. To set the format flags of the output field

**Answer: C**

9. **What is the purpose of the `std::setprecision()` function?**

- A. To set the width of the output field
- B. To set the precision of the output field
- C. To set the fill character of the output field
- D. To set the format flags of the output field

**Answer: B**

10. **Which header file must be included to use user-defined manipulators in C++?**

- A. `<string>`
- B. `<fstream>`
- C. `<iostream>`
- D. `<iomanip>`

**Answer: D**

## Lec 39 - Pointers

1. **What is the operator used to declare a pointer in C++?**

- A) &
- B) \*
- C) \$
- D) %

**Answer: B**

2. **What is the difference between a null pointer and a void pointer in C++?**

- A) A null pointer points to nothing, while a void pointer points to any data type.
- B) A null pointer points to a memory address, while a void pointer points to a function.
- C) A null pointer is used for deallocation, while a void pointer is used for allocation.
- D) A null pointer is used for function pointers, while a void pointer is used for object pointers.

**Answer: A**

3. **What is the correct syntax for dynamically allocating memory for a pointer in C++?**

- A) `int* p = malloc(sizeof(int));`
- B) `int* p = new int;`
- C) `int p = new int;`
- D) `int* p = malloc(int);`

**Answer: B**

4. **What is a dangling pointer in C++?**

- A) A pointer that points to a valid memory address.
- B) A pointer that points to a deallocated memory address.
- C) A pointer that points to a null memory address.
- D) A pointer that points to a function.

**Answer: B**

5. **What is the purpose of the const keyword when working with pointers in C++?**

- A) To declare a constant pointer variable.
- B) To declare a pointer to a constant variable.
- C) To declare a constant memory address.
- D) To declare a constant value pointed to by a pointer.

**Answer: B**

6. **What is the difference between a reference and a pointer in C++?**

- A) A reference is an alias for a variable, while a pointer is a separate variable that stores a memory address.
- B) A reference is a separate variable that stores a memory address, while a pointer is an alias for a variable.
- C) A reference and a pointer are the same thing in C++.
- D) A reference is used for dynamic memory allocation, while a pointer is used for static memory allocation.

Answer: A

7. What is the operator used to access the value pointed to by a pointer in C++?

- A) .
- B) ->
- C) &
- D) \*

Answer: D

8. What is the correct syntax for deleting a dynamically allocated pointer in C++?

- A) delete p;
- B) delete \*p;
- C) delete &p;
- D) free(p);

Answer: A

9. What is a memory leak in C++?

- A) A pointer that points to a deallocated memory address.
- B) A pointer that points to a null memory address.
- C) A pointer that points to a valid memory address.
- D) A failure to deallocate dynamically allocated memory, causing the program to use up all available memory.

Answer: D

10. What is a smart pointer in C++?

- A) A pointer that automatically deallocates memory when it goes out of scope.
- B) A pointer that automatically allocates memory when it is assigned a value.
- C) A pointer that automatically sets the value pointed to by the pointer to null.
- D) A pointer that automatically dereferences itself when used in code.

Answer: A

## Lec 40 - Objects as Class Members

### 1. What is composition in C++?

- A. The process of defining classes
- B. The process of defining objects as class members
- C. The process of creating pointers
- D. The process of using inheritance

Answer: B

### 2. What is the purpose of defining objects as class members?

- A. To create complex data structures
- B. To access private data members of another class
- C. To define a new data type
- D. To implement inheritance

Answer: A

### 3. How are objects as class members constructed and destructed?

- A. They are never constructed or destructed
- B. They are constructed and destructed along with the parent object
- C. They are constructed and destructed separately from the parent object
- D. They are only destructed when the parent object is destructed

Answer: B

### 4. How do you access an object that is a member of a class?

- A. Using the -> operator
- B. Using the :: operator
- C. Using the dot (.) operator
- D. Using the \* operator

Answer: C

### 5. Which relationship can be implemented using objects as class members?

- A. "is-a"
- B. "has-a"
- C. "inherits-from"
- D. "contains-a"

Answer: B

### 6. Can an object be a member of more than one class?

- A. Yes, but only if both classes are derived from the same base class
- B. No, an object can only be a member of one class
- C. Yes, an object can be a member of any number of classes
- D. Yes, but only if the classes are in the same namespace

Answer: B

7. **What happens if you try to assign one object to another object that is a member of a class?**
- A. The program crashes
  - B. A compiler error is generated
  - C. The object is copied to the new object
  - D. The object is deleted and a new object is created

**Answer: C**

8. **Can you use objects as class members in a union?**
- A. Yes, but only if the objects are of the same type
  - B. No, objects cannot be used as class members in a union
  - C. Yes, objects can be used as class members in a union
  - D. Yes, but only if the union is in a namespace

**Answer: B**

9. **Can you use pointers to access objects that are members of a class?**
- A. Yes, but only if the objects are public
  - B. Yes, but only if the objects are static
  - C. No, pointers cannot be used to access objects that are members of a class
  - D. Yes, you can use pointers to access objects that are members of a class

**Answer: D**

10. **How do you initialize an object that is a member of a class?**
- A. Using the new operator
  - B. Using the sizeof operator
  - C. Using a constructor
  - D. Using a destructor

**Answer: C**

## Lec 41 - Template Functions

### 1. What is a template function?

- a) A function that only works with one specific data type
- b) A function that can work with multiple data types
- c) A function that is used to create classes
- d) A function that is only used in object-oriented programming

Answer: b

### 2. What is the syntax for declaring a template function in C++?

- a) `template<typename T> void functionName(T arg);`
- b) `void functionName<T>(T arg);`
- c) `template<T> void functionName(T arg);`
- d) `typename T void functionName(T arg);`

Answer: a

### 3. What is the purpose of a template function?

- a) To create a specialized function for a specific data type
- b) To create a function that can work with multiple data types
- c) To create a function that is used for input/output operations
- d) To create a function that is used for debugging purposes

Answer: b

### 4. How does a template function differ from a regular function?

- a) A template function can only work with one data type
- b) A template function cannot be called directly
- c) A template function can work with multiple data types
- d) A template function does not need a return type

Answer: c

### 5. What is a template parameter?

- a) A variable used to hold data
- b) A type used to represent a data type in a template function
- c) A function used to manipulate data
- d) A value that is returned by a function

Answer: b

### 6. Can template functions be overloaded?

- a) Yes
- b) No

Answer: a

### 7. What is a template specialization?

- a) A way to create a specialized version of a template function for a specific data type

- b) A way to create a template function that works with all data types
- c) A way to create a function that cannot be used with templates
- d) A way to create a function that can only be used with templates

**Answer: a**

**8. What is a non-type template parameter?**

- a) A variable used to hold data
- b) A type used to represent a data type in a template function
- c) A value used to represent a constant in a template function
- d) A function used to manipulate data

**Answer: c**

**9. Can templates be used with classes?**

- a) Yes
- b) No

**Answer: a**

**10. What is the benefit of using template functions?**

- a) Reduced development time
- b) Increased compilation time
- c) Limited functionality
- d) Limited reusability

**Answer: a**

## Lec 42 - Class Templates

### 1. What is a class template in C++?

- a. A class that is used to create multiple instances of a class
- b. A class that can be used with multiple data types
- c. A class that is used to define a single data type
- d. A class that cannot be instantiated

Answer: b

### 2. How do you declare a class template in C++?

- a. By using the keyword "class" followed by the template parameter list and the class declaration
- b. By using the keyword "template" followed by the template parameter list and the class declaration
- c. By using the keyword "typedef" followed by the template parameter list and the class declaration
- d. By using the keyword "typename" followed by the template parameter list and the class declaration

Answer: b

### 3. What is the purpose of a template parameter in a class template?

- a. To represent a data type that can be used by the class
- b. To represent a constant value that can be used by the class
- c. To define the size of the class
- d. To determine the access level of the class members

Answer: a

### 4. How do you instantiate a class template in C++?

- a. By defining a new class with the same template parameters as the original class
- b. By creating an object of the class with the desired data type as the template argument
- c. By using the keyword "using" followed by the name of the class template
- d. By using the keyword "typename" followed by the name of the class template

Answer: b

### 5. How does template specialization work in class templates?

- a. It allows you to create a specialized version of the class for a specific data type or value
- b. It allows you to define a new template parameter for the class
- c. It allows you to restrict the data types that can be used with the class
- d. It allows you to override the default implementation of the class

Answer: a

### 6. How do you define member functions for a class template in C++?

- a. By defining the functions outside the class definition
- b. By defining the functions inside the class definition
- c. By using the keyword "template" before the function definition

d. By using the keyword "typename" before the function definition

Answer: b

7. **How do you overload a class template in C++?**

- a. By defining a new class with the same name but different template parameters
- b. By defining a new member function with the same name but different template parameters
- c. By defining a new member function with a different name but the same template parameters
- d. By defining a new class with a different name but the same template parameters

Answer: b

8. **What are the advantages of using class templates in C++?**

- a. They provide code reusability and improve code quality
- b. They reduce development time and cost
- c. They allow for generic programming and flexible data structures
- d. All of the above

Answer: d

9. **What are the potential drawbacks of using class templates in C++?**

- a. They can lead to longer compilation times
- b. They can be difficult to understand for novice programmers
- c. They can be prone to errors and bugs
- d. All of the above

Answer: d

10. **Can class templates be used with user-defined data types?**

- a. Yes, as long as the data type is specified as a template parameter
- b. No, class templates can only be used with built-in data types
- c. It depends on the complexity of the user-defined data type
- d. It depends on the implementation of the class template

Answer: a

## Lec 43 - Programming Exercise - Matrices

1. **What is a matrix in programming?**

- A. A tool used for debugging code
- B. A rectangular array of numbers
- C. A type of conditional statement
- D. A data structure used for storing strings

**Answer: B. A rectangular array of numbers**

2. **What is the process of adding two matrices called?**

- A. Multiplication
- B. Subtraction
- C. Addition
- D. Division

**Answer: C. Addition**

3. **Which data structure is commonly used for representing matrices in programming?**

- A. Stack
- B. Queue
- C. Array
- D. Linked list

**Answer: C. Array**

4. **What is the result of multiplying a 3x2 matrix with a 2x3 matrix?**

- A. A 3x3 matrix
- B. A 3x2 matrix
- C. A 2x3 matrix
- D. A 2x2 matrix

**Answer: A. A 3x3 matrix**

5. **What is the identity matrix?**

- A. A matrix with zeros in all its elements
- B. A matrix with ones in all its elements
- C. A matrix with zeros in all its diagonal elements and ones in all its other elements
- D. A matrix with ones in all its diagonal elements and zeros in all its other elements

**Answer: D. A matrix with ones in all its diagonal elements and zeros in all its other elements**

6. **Which of the following is used for finding the determinant of a matrix?**

- A. Gaussian elimination
- B. LU decomposition
- C. QR decomposition
- D. Singular value decomposition

**Answer: A. Gaussian elimination**

7. **Which of the following is true about a symmetric matrix?**

- A. It has equal number of rows and columns
- B. It is a square matrix
- C. It is equal to its transpose
- D. It has only positive numbers as its elements

**Answer: C. It is equal to its transpose**

8. **What is the inverse of a matrix?**

- A. A matrix with all its elements multiplied by -1
- B. A matrix with all its elements squared
- C. A matrix that when multiplied by the original matrix gives the identity matrix
- D. A matrix with all its elements equal to the reciprocal of the original matrix

**Answer: C. A matrix that when multiplied by the original matrix gives the identity matrix**

9. **Which of the following operations is not possible with matrices?**

- A. Addition
- B. Subtraction
- C. Multiplication
- D. Division

**Answer: D. Division**

10. **Which of the following is used for solving systems of linear equations represented by matrices?**

- A. Gaussian elimination
- B. LU decomposition
- C. QR decomposition
- D. Singular value decomposition

**Answer: A. Gaussian elimination**

## Lec 44 - Matrix Class

### 1. What is a matrix class in programming?

- A) A data structure that represents a collection of integers
- B) A programming construct that encapsulates the properties and behaviors of matrices
- C) A set of mathematical functions for manipulating matrices
- D) A type of programming language syntax

Answer: B

### 2. What kind of operations are typically included in a matrix class?

- A) String concatenation and splitting
- B) Looping and branching
- C) Matrix addition, multiplication, transposition, and determinant finding
- D) File I/O and network communication

Answer: C

### 3. Why is a matrix class useful in programming?

- A) It simplifies the implementation of matrix operations in programs
- B) It allows programmers to manipulate matrices with ease
- C) It promotes code reuse and modularity
- D) All of the above

Answer: D

### 4. What are some typical member variables of a matrix class?

- A) Dimensions and element values
- B) String and integer values
- C) Boolean and float values
- D) Time and date values

Answer: A

### 5. What is the purpose of encapsulation in a matrix class?

- A) To hide the implementation details of the class
- B) To allow external access to the class's member variables
- C) To expose the class's internal workings to other classes
- D) None of the above

Answer: A

### 6. Which of the following is an example of a matrix operation that can be performed in a matrix class?

- A) Sorting the elements of a matrix in ascending order
- B) Removing duplicate elements from a matrix
- C) Transposing a matrix
- D) Merging two matrices into one

Answer: C

7. **Which of the following is a benefit of using a matrix class in programming?**

- A) It can make programs more efficient by optimizing matrix operations
- B) It can help catch errors in matrix calculations
- C) It can make programs easier to read and understand
- D) All of the above

**Answer: D**

8. **How do matrix classes differ from other programming constructs?**

- A) They are a type of loop construct
- B) They are a type of branching construct
- C) They are a type of data structure
- D) They are a type of function

**Answer: C**

9. **Which of the following is an example of a matrix class method for accessing the elements of a matrix?**

- A) `get_element()`
- B) `add_element()`
- C) `delete_element()`
- D) `count_elements()`

**Answer: A**

10. **Which of the following is an example of a matrix class method for modifying the elements of a matrix?**

- A) `get_element()`
- B) `add_element()`
- C) `delete_element()`
- D) `set_element()`

**Answer: D**

## Lec 45 - Example (continued)

1. **What does "example (continued)" refer to in programming documentation?**

- A) A new example
- B) A continuation of a previously mentioned example
- C) An unrelated example
- D) None of the above

**Answer: B**

2. **When is "example (continued)" typically used in programming documentation?**

- A) To introduce a new code snippet
- B) To provide further details on a previously mentioned code example
- C) To demonstrate an unrelated feature
- D) None of the above

**Answer: B**

3. **How can "example (continued)" help developers in programming documentation?**

- A) It can provide further clarification on how to implement a specific feature or function
- B) It can improve their overall comprehension of the code
- C) Both A and B
- D) None of the above

**Answer: C**

4. **What is the purpose of including code examples in programming documentation?**

- A) To demonstrate how to use a particular feature or function
- B) To provide a practical understanding of the code
- C) Both A and B
- D) None of the above

**Answer: C**

5. **What is the benefit of using "example (continued)" in programming documentation?**

- A) It can make the code more organized and easier to read
- B) It can help developers better understand the code
- C) Both A and B
- D) None of the above

**Answer: C**

6. **When should "example (continued)" be used in programming documentation?**

- A) Whenever there is a code example
- B) Only when there is a need to provide further details on a previously mentioned code example
- C) Whenever the programmer wants to show off their coding skills
- D) None of the above

**Answer: B**

7. **What is the purpose of encapsulation in programming?**

- A) To hide implementation details of a class or function
- B) To promote code reuse and modularity
- C) Both A and B
- D) None of the above

**Answer: C**

8. **Which programming principle allows the programmer to use the same operator symbols (+, \*, etc.) to perform matrix operations as they would for regular arithmetic operations?**
- A) Encapsulation
  - B) Inheritance
  - C) Operator overloading
  - D) Polymorphism

**Answer: C**

9. **What is the main benefit of using a matrix class in programming?**
- A) It allows for easy manipulation and analysis of complex data sets
  - B) It can help solve real-world problems in fields like engineering and finance
  - C) Both A and B
  - D) None of the above

**Answer: C**

10. **Which programming approach is typically used to implement a matrix class?**
- A) Functional programming
  - B) Object-oriented programming
  - C) Procedural programming
  - D) Declarative programming

**Answer: B**

