CS201

SHORT NOTES FOR FINAL TERM

PROVIDE BY ORANGE MONKEY TEAM

LECTURE 23 TO 25

Pre-processor:

Macros with the #define directive. Micro is a special name, which is substituted in the code by its definition and as a result, we get an expanded code.

#define PI 3.1415926

Directive:

Directive is a language construct that specifies how a compiler should process its input

Macro:

Macro is a fragment of code which has been given a name.

Memory Allocation:

Memory allocation refers to performing manual memory management for dynamic memory allocation in the C programming language via a group of functions in the C standard library realloc, calloc etc.

Dangling Pointers:

Dangling pointer occurs at the time of the object destruction when the object is deleted or de-allocated from memory without modifying the value of the pointer.

History of C/C++:

C language was developed by scientists of Bell Labs in 1970s. It is very lean and mean language, very concise but with lot of power. C conquered the programming

world and took it by storm. Major operating systems e.g., Unix was written in C language.

Structured programming:

C is called structured modular programming language because while solving large and complex problem, C programming language divides the problem into smaller modules called functions.

Inline functions:

This is also one of the facilities provided by C++ over C. In our previous lectures, we discussed and wrote macros few macros like max and circlearea. Inline functions also work more or less in the same manner as macros.

LECTURE 26 TO 45

Q.1. What is OOP(Object Oriented Programming)

Ans. Object Oriented Programming is a computer programming model that organizes software around data or subject rather than features or logic. An **object** can be defined as a data that has a unique attribute and behavior

Q.2. What is Core?

Ans. The core of the pure object oriented programming is to create an object. In code, that has certain properties and method.

Q.3. What is class?

Ans. A class is a user defined data type, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class

E.g: Consider the class of Cars

Q.4 How many types of class?

Ans. There are two types of classes

1. Abstract classes
2. Concrete Classes

Q.5. What is Abstract Classes?

Ans. Not object is made, Only inheritance can be used that have abstract and concrete methods, real words existence are not found. Abstract class can not be instantiated using new keyword. Abstract class may or may not have abstract method

Q.6. What is concrete methods?

Ans. Object is made and used, derived and child classes from abstract classes , real words existence are found. Concrete class can be instantiated using new keyword. Concrete class can not have an abstract method

Q.7. What is abstraction or Information hiding?

Ans. Data abstraction refers to provide essential information to the outside the world but hide their background details. i.e to represent the information needed to the program without presentation the details

For Example: The data base system hides detials how data is stored and created and maintained.

Q.8 What is Encapsulation?

Ans. Encapsulation is placing the data and function work on that data in the same place. Only works with procedural languages.

It is not always clear which works on which variables but object oriented programming provides you framework to place the data that works together in the same object.

Q.9 What is inheritance?

Ans. One of the most useful aspects of object oriented programming is code reusability. Inheritance is the process of forming a new class from an existing class that is:

New class is called as derived class and **Existing class** is called as base class.

Q.10. What is the features to reduce the code size in OPP?

- Public inheritance
- Protected
- Private

Q.11. What is Polymorphism?

Ans. Poly refers in many. This is a single function or an operator functioning in many ways different upon the usage is called Polymorphism.

Q.12. What is overloading?

Ans. Is the branch of polymorphism. When existing function is made to operate on new date type, it is said to be overloading

- Function overloading
- Operator overloading
- Overloading vs overloading

Q.13. What is constructor?

Ans. We have written a function named Date(int, int, int). This is in the public section in our class. It has no return type having the name as that of class. Such function are called constructor.

Q.14. What is friend function?

Ans. A friend function that can access protected, private and public member of a class. The friend function is **declared** using friend keyword inside the body of a class.

Q.15. What is friend class?

Ans. Friend class is a class that can access to private and protected member of a class which is declared as friend.

For example: This is needed we want to allow particular class to access the private and protected member of a class.

Q.16. What is reference data type?

Ans. Unlimited numbers of reference data types, as they are defined by the user. Consists of Boolean and numeric data type, Int, float, long, short and double

Example of reference data types are Arrays, Sting Class etc.

Q.17. What is default arguments with constructors?

Ans. A constructor can have two arguments. They are used to initialize member objects. If default values are supplied, the trailing arguments can be omitted in the expression list of the constructor.

Example: Defines a class with one constructor and two default constructor.

Q.18. What is operator overloading and Explain its type?

Ans. Operator overloading is quite similar to function overloading.

There are two types of Operator overloading:

A) Unary

B) Binary

Unary: Unary operator requires only ones to the left of the operand on single operator.

Example: ^, &, !

Binary: Binary operator requires two operands on both side of the operator.

Example: +, -, *, /

Q.19. Write impact of restrictions on operator overloading?

- Semantics(Behavior) cannot be changed
- Arity(Number of operands) cannot be changed
- Precedence(High/low precedence) cannot be changed
- Associativity(left/right) of the operation cannot be changed

Q.20. What is Non-member Operator Functions?

Ans. Operator can be written as Non-member without making them as friend. But this is radians and less efficient way therefore, it is not recommended.

Q.21. Write the set of Rules of conversation operator?

Ans. Rules are in the following below:

- When assigning a value, e.g if you assign an integer to a variable of type long, the compiler converts the integer to a long.
- When performing an arithmetic operation, For example, if you add an integer and a floating point value the compiler converts the integer to a float before it performs the addition.
- When passing an argument to a function, E.g if you pass an integer to a function that expects a long.
- When remaining value from a function, E.g if you return a float from a function that has double as its return type.

In all of these situations, the compiler performs the conversion implicitly. The conversion explicit is possible by using a cast expression.

Q.22. Can we do conversation with objects of our own classes?

Ans. Yes, if we go to the basic definition of a class, it is nothing but a user defined data type. A class is a user defined data type, we can also define conversion on it.

Q.23. What is constructor and destructor and global array?

Ans. **New** is called before the constructor and **Delete** is called after the destructor.

A **global array** called pool that can store all the Name object expected.

Q.24. What is meant by copying a file in the reverse order?

Ans: Copying a file in the reverse order means the last byte of the input file will be the first byte of the output file, second byte of the input file will be the second

byte of the output file until the first byte of the input file becomes the last byte of the output file.

Q.25. What is Stream in C++?

Ans. A C++ stream is a flow of data into or out of program such as the data written to cout or read from cin.

Cin is an example of istream. ostream is a general purpose output stream. Every stream has an associated source and a destination.

Q.26. Write Predefined stream Objects?

- cin_____ Standard input
- Cout_____ Standard output
- Cerr_____ Standard error with unbuffered output.
- Clog_____ Standard error with buffered output
- Caux_____ Auxiliary (DOS only)
- Cprn_____ Printer(DOS only)

All these functions (getline, get read, unget and peek) are implemented as member function of the input class.

Q.27. What is stream manipulators?

Ans: Stream Manipulators are functions specifically designed to be used in conjunction with the insertion(<<) and extraction(>>) operators on stream objects.

E.g std::cout<<std::setw(10)

Q.28. What is manipulators?

Ans. Manipulators are helping functions that can modify the input/output stream. It does notnot mean that we change the value of variable, it modifies the I/O stream using insertion(<<) and extraction(>>) operators.

Q.29. What is Non-Parameterized Manipulators and Parameterized manipulators?

Ans. Non-parameterized manipulators do not take argument to control the formatting of input/output

Parameterized manipulators take argument for formatting.

Q.30. What is overloading insertion and extraction operators?

Ans. Insertion operator(<<) is used for output and extraction operators(>>) is used for input. These operators must be overloaded as a global function.

- operator << returns a value of type ostream & and operator >> returns a value of type istream & to support cascaded operations.
- For operator >>, the second parameter must also be passed by reference.
- Stream insertion (<<) and extraction operators (>>) are always implemented as non-member functions.
- The second parameter to operator << is an object of the class that we are overloading the operator for. Similar is the case for operator >>.

Q.31. What is user defined manipulators?

Ans. Manipulators are special stream functions that are used to change certain characteristics of the input and output.

Q.32. What is static keyword and Static objects and Static data members of a class?

Ans. In java, **static keyword** is mainly used for memory management. It can be used with variables, methods, blocks and nested classes. It is a keyword which is used to share same variable or method of a given class.

- **Static objects in java** the variable or function is shared between all instances of that class, not the actual objects themselves.
- **Static data member** are class members that are declared using static keyword. A static member has certain special characteristics.

Q.33. What is a Pointer?

Ans: Pointer is a special type of variable that contains a memory address. It is not a variable that contains a value rather an address of the memory that is contained inside a pointer variable.

Q.34. What is references?

Ans: A reference can be considered as a special type of pointer as it also contains memory address.

Q.35. What is call by value?

Ans. The call by value method of passing arguments to a function copies the actual value of an argument into the formal parameter of the function.

Q.34. What is call by reference?

Ans. The call by reference method of passing arguments to a function copies the address of an argument into the formal parameter. It means the changes made to the parameter affect the passed argument.

The use of call by reference is also important for the sake of efficiency

Q.35. What is dynamic memory allocation?

Ans: Dynamic memory allocation refers to managing system memory at runtime.

- malloc()
- calloc()
- realloc()
- free()

Q.36. What is assignment and initialization?

Ans: **Initialization** gives a variable an initial value at the point when it is created.

Assignment a variable a value at the some point after the variable is created.

Q.37. What is copy constructor and use?

Ans. A copy constructor is a constructor that creates an object using another object of the same java class.

Usage: we want to copy a complex object that has several fields. We want to make a deep copy of an existing object.

Q.38. What is the object as class members and its advantages?

Ans. In object- oriented programming, a member variable is a variable that is associated with a specific object, and accessible for all its methods.

Advantages:

- Objects can Encapsulate Operations along with data.
- Objects are efficient
- Objects can represent Part-whole relationship

Q.39. What is class templates and Non-type parameters?

- **Class template** can make use of another kind of template parameter known as non-type parameter.
- **Non-type parameter** is a special type of parameter that does not substitute for a type, but is instead replaced by a value.

Q.40. What is templates and friend function?

Ans. A template function instantiated with one set of template arguments may friend to one template class instantiated with the same set of template arguments.

Q.41. What is programming Exercise- matrices and design recipe and program analysis?

Ans. Mathematics is a good domain to develop different classes and programs. E.g solutions for complex numbers, Matrices and Quadratic equations can be sought for developing our own classes.

- **Design recipe** is a roadmap for defining function, which programmers use to make sure code they write does what they want it to do.
- **Program analysis** is the process of looking at an organization's intended social and behavior change communication program and then identifying enabling.

Q.42. What is Matrix class and matrix constructor?

Ans. A **matrix** can have a number of rows and columns and each cell of a matrix can be represented by their combination

A **matrix constructor** is a special 'member function having the same name as that of its class which is used to initialize some valid values to the data members of an object.

Q.44. What is destructor of Matrix class?

Ans: A destructor is called for a class object. Destructor are usually used to de-allocate memory and do other cleanup for a class object and its class members when the object is destroyed.

Q.45. What is array and loop and variables?

- An **array** is a type of data structure. We use an array to store multiple values of the same data type
- A **loop** is a sequence of instructions that is continually repeated until a certain condition is reached.
- A **variable** is a name given to a memory location.

Q.46. What is addition operator function and minus operator function and multiplication operator function?

- **Addition assignment operator**(+=) adds the value of the right operand to a variable and assigns the result to the variable.
- **Minus assignment operator** minus the value right operand from the left operand and assigns the result to the left operand.

- **Multiplication assignment** operator multiplies the right operand with the left operand and assigns the result to the left operand.

Q.47. What is an assignment operator?

Ans. Assignment operators are used to assigning value to a variable.

Q.48. What is input function and transpose function?

Ans. A utility function transforms an outcome into a numerical value and measure the worth of an outcome.

Two types of utility function:

1. Input function
2. Transpose function

- **Transpose function** returns a vertical range of cells as a horizontal range or vice versa.
- **Input function** is used to ask the user of the program a question, and then wait for a typed response

Most useful operator function in C++ for mcqs

Operator **Meaning**

- =_____Assignment
- +_____Concatenation
- +=_____Concatenation assignment
- ==_____Equality
- !=_____Inequality
- <_____Greater than
- <=_____Greater than or equal to
- >_____less than
- >=_____ less than or equal to
- >>_____ Reads
- >>_____Prints

Header files	**full forms**
iostream.h	Include all input and output.streams
conio.h	All console input and output functions
math.h	include all mathematical functions
stdllb.h	include all standard library functions
string.h	All string manipulation functions
ctype.h	all character manipulating function
iomanip.h	include all input and output manipulators
ftype.h	include all file type function