|  |  |
| --- | --- |
| Day 1 | * Source code vs Object Code   Source code is generally referred to the programming statements written by programmer, while the object code refers to the output file or compiled file produced after compilation of source code. |
|  | * Using Compiler with flag   Different flags can be used during file compilation like –verbose, -d, etc. |
|  | * Keywords and variables   Keywords are the reserved/predefined words, ex. static, final, etc.  Variables are declared by programmer to store data. |
|  | * Variable Scope, lifetime, initial value, modifiers   Scope of the variable is determined at compile time. If the variable is declared in class privately, then it can’t be accessed outside. Lifetime of variable is the time during which variable stays in the memory. Default value of variable depends on the datatype, ex: for int initial value is 0. For variable access modifiers used are public, private and protected. |
|  | * Programming constructs (Selection construct and loop constructs)   Selection construct is basically the decision to be made by program based on previous results.  Loop construct is repeating the execution for section of code. |
|  | * Declare array   For declaring array, first need to choose data type and then allocate memory to array using new keyword. |
|  | * Store data in array and process elements   In array data is stored sequentially (contiguous memory allocation). The elements can be accessed using index. |
| Day 2 | * Pointers, Declare and Initialize pointers   Pointers are used to store address of memory used by variables. |
|  | * Function definition/invocations   Method is a collection of statements that perform specific task and returns the results on invocation. |
|  | * Call function by passing value or address   Calling function by passing value means copy of the parameter is made in memory. |
|  | * Dynamic memory allocation/de allocation   Dynamic memory is allocated during run time. Heap offers APIs to allocate as well as deallocate memory for objects. Garbage collection is called to destroy objects which are not used by java applications anymore. |
|  | * Memory leakage   It happens when memory which is no longer needed is not deallocated/released. |
|  | * User Defined Structures   User defined structures creates single data-type that represents more than one built-in data type. |
|  | * Access Data members   Data members declared as public are accessible from anywhere in program. And the data declared as private can be used by creating instance of the class. |
|  | * Dynamic Memory Allocation for user defined structure   In Java, for all objects, memory is dynamically allocated in heap. |
|  | * Array of pointers   It is a set of variables, where variables are pointers. |
|  | * Iterate on array of pointers   Array of pointers can be iterated in the same way we would iterate any array. |
| Day 3 | * Implement Data Structures   In Data structures, we can store data of same data types. |
|  | * Linked List   In Linked list, data is stored in non-contiguous manner. Dynamic memory allocation. Data can be accessed sequentially. For accessing data pointers are used. |
|  | * Stack   In stack, data is accessed sequentially. Last In First Out policy is used for insertion and deletion operations. |
|  | * Queue   In queue, memory allocation is done dynamically, Data can be accessed sequentially. First In First out policy is used for inserting and deleting. |
|  | * Binary Tree   Binary tree is a hierarchical data structure. It has at most two children. |