1.	**algorithm**	Step by step process.
2.	**AND**	The AND operation is a binary operation, meaning that it needs two operands. c = a AND b Both a AND b must be true for the result to be true. Example: if (num1 > 0 && num1 < 100) // && indicates AND
3.	**Boolean logic**	Boolean logic is a form of mathematics in which the only values used are true and false. Boolean logic is the basis of all modern computing. There are three basic operations in Boolean logic - AND, OR, and NOT.
4.	**Class**	A template of an object. Objects are instantiated from a class.
5.	**Class**	The blueprint that defines the obejcts with the properties and the methods
6.	**Computer program**	a step by step set of instructions for a computer.
7.	**Constructor**	A method that has the same name as the class, but it does not have a return value specified. A constructor builds the object or class structure in memory.
8.	**Copy Constructor**	passes as a parameter values anouther object. All instance variables in the object
9.	**Default constructor**	Does not take any input values this constructor assigns default initial values to all instance variables.
10.	**difference between Static and non Static**	A "Static" method DOES NOT require instantiating an object to access it. Examples : Main and MATH methods A "Non Static" method requires instantiating an object to access it.
11.	**Element and index**	...
12.	**Event-Driven Programing**	An event occurs whenever an event listener detects an event trigger and responds by running a method called an event handler.
13.	**Events**	allow components to notify each other when something happens.
14.	**Getter methods**	the methods that retrieve a property's value are called getter methods. getSize()
15.	**GUI**	Graphical User Interfaces A GUI has icons on the computer screen and a mouse (or other device) to control a pointer that can be used to operate the computer.
16.	**Inheritance**	allows code defined in one class to be reused in other classes.
17.	**Instance**	each copy of an object from a particular class is call an instance of the object.
18.	**Instantiation**	The act of creating a new instance of an object.
19.	**Iteration**	Is another name for looping. An iterative method contains a loop.
20.	**Listeners**	The objects that receive the information that an event occurred are called listeners.
21.	**Methods**	In object-oriented programming, the programs that manipulate the properties of an object are the object's method
22.	**NOT**	The NOT operation is a unary operation with only one operand. c = NOT (a) It simply reverses the true or false value of the operand. Example: while (!exit) // ! Indicates NOT
23.	**Object**	anything that can be represented by data in a computer's memory and manipulated by a computer program.
24.	**Object Oriented**	you can define objects using the Class construct. Example: JAVA, C++
25.	**OR**	The OR operation is also a binary operation with two operands. c = a OR b If either a OR b is true, then the result is true. Example: if(num1<=0 || num1>= 100) // || indicates OR
26.	**Parameter(overloaded) constructor**	passes values through the parameter list; these values are used to initialize all instance variables.
27.	**Parameters**	Are values that are passed to another method when it is invoked by an event.
28.	**Procedural**	A Procedural Language does NOT PERMIT to define OBJECTS. Example: C Language, PASCAL

29.	**Program development Cycle**	· Design a solution to a problem (design a program) · implement the solution (code the program) · Test the solution (test the program) · Fix the solution (debug the program)
30.	**Properties**	The data that represent the object and organized into a set of properties.
31.	**reading a file**	: Open file, Priming read, Loop until EOF, second read in loop and Close file after loop.
32.	**Setter methods**	The methods that change a property's value are called setter methods setSize()
33.	**State**	The values stored in an object's properties at any one time form the state of an object.
34.	**Type of errors**	SYNTAX ERRORS - Violations of the programming language rules. LOGIC ERRORS - Also called run-time or execution errors. They are errors in the sequence of the instructions in the program.
35.	**Variables**	Variables are memory locations use to store data. Methods maintained their own variables which are local to the methods