

AP Computer Science A Study Guide Unit 3

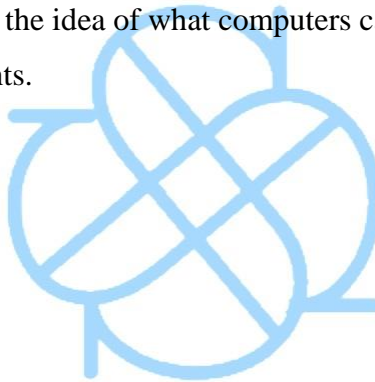
From Simple Studies: <https://simplestudies.edublogs.org> &

@simplestudiesinc on Instagram

Boolean Expressions

Boolean Expressions

- Boolean logic: answers can be true or false (no middle ground)
- George Boole
 - Inventor of the concept of Boolean logic
 - He also developed the idea of what computers can use to make decisions – condition statements.



if Statements and Control Flow

- An `if` statement tells the computer that if a condition is true, then it should execute the block of code within the function.
 - Most basic control flow statement
 - If it's false, the computer should skip the code and continue with the rest of the program.
- Control flow statements
 - Can modify or break the flow of the execution
 - Implementing decision making
 - Looping
 - Branching program
 - To execute certain blocks of code in your program.
- Switch statements
 - Can test a range of values
 - Can use these statements if the if-else chain is too long.
- Three types of control flow supported by Java
 - Decision making statements
 - `if – then`
 - `if – then – else`
 - Looping statements
 - `for`
 - `while`
 - `do-while`
 - Branching statements
 - `break`
 - `return`
 - `continue`

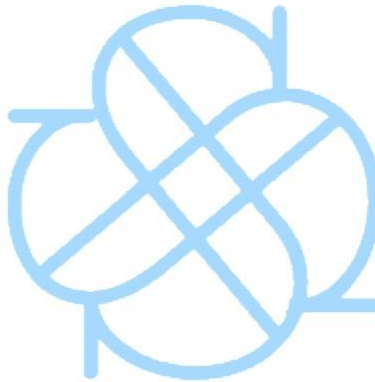
if-else Statements

- `if-else` statements

- If the condition is true, the computer executes the block of code but if the condition is false, the computer executes the else action.
- Will always take some action because the condition has to be either true or false (not in the middle).
- Optional else statement
 - Alternate path if the boolean value might be false

else-if Statements

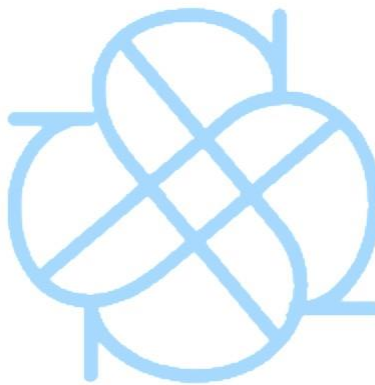
- `if-else-if` statements
 - In Java, once a block of code is executed, it doesn't check for the other else-if conditions
 - Breaks the loop



Equivalent Boolean Expressions

- De Morgan's Laws
 - $\text{not } (a \text{ and } b) \rightarrow (\text{not } a) \text{ or } (\text{not } b)$
 - $\text{not } (a \text{ or } b) \rightarrow (\text{not } a) \text{ and } (\text{not } b)$

- How De Morgan's Laws are expressed in programming languages!
 - $!(a \ \&\& \ b)$ is equivalent to $!a \ || \ !b$
 - $!(a \ || \ b)$ is equivalent to $!a \ \&\& \ !b$
- Negating boolean expressions
 - You can do this with relational operators
 - $<$, $>$, or $==$
- Truth Tables are used to prove boolean expressions.
- Equivalent boolean expressions result in the same value in any case.



Comparing Objects

- Object – characterized by state, attributes, and behavior.
 - Instance of a class
- Class – Software blueprint
 - Implement Objects of a specific data type
- Methods

- Provide behaviors of the object
 - Provide operations which manipulate the object
- Comparing String Objects
 - Using the equals method
 - If (string1.equals(string2))...
 - Using the compareTo method
 - Int compareTo (string otherStringYouAreComparing)
 - stringOne. compareTo (stringTwo) < 0
 - stringOne precedes stringTwo in a dictionary
 - stringOne. compareTo (stringTwo) > 0
 - stringTwo precedes stringOne in a dictionary
 - stringOne. compareTo (stringTwo) == 0
 - both Strings are identical
 - String Objects – immutable
 - No methods can change them after they are constructed
 - You can create a new String object which can be a modified form of an existing String object.
- Characters in the String objects
 - Compared based on their position in the ASCII chart.
- Don't use (==) for testing or comparing String Objects.