Assignment 2

Binary Decision Diagram

In this assignment I have implemented Reduced ordered Binary Decision Diagram – BDD. My BDD will take as an input a string containing a Boolean function and create a tree using Shannon decomposition of the given Boolean function. After creation it will return the root node of the tree.

Tree body

Text

Description automatically generated

This is my constructor of the BDD tree. It contains root of the tree. Order of Boolean arguments that determine how the tree will be created using Shannon decomposition. A dictionary – used\_nodes – where all unique nodes are stored. Number of unique nodes and number of unique Boolean function arguments.

Node body

Text

Description automatically generated

Every node that is created contains its parent, pointer to left\_child that will contain lower node and pointer to right\_child that will contain higher node based on Shannon decomposition. Every node also contains number of unique variables in current function. Variable count is later used in function .use() where we traverse the tree.

Tree function: create(„Boolean function“, [order of arguments])

How to use the create function:



Firstly you need to create a variable that will be an object BDD(). Then you can call the function use and insert arguments. Arguments for boolean function needs to only contain uppercase Alphabet letters, ! indicating negation of the boolean variable. If you want to multiple variables just place them next to each other without any kind of character.

Example: „AB+C+!DE+!F!G“

After you create the correct boolean function, the second argument is list of the order of inserted boolean variables for Shannon decomposition.

Example: [‚A‘,‘B‘,‘C‘,‘D‘,‘E‘,‘F‘,‘G‘]