ASSIGNMENT 2: CONSTRUCTING CONTROL FLOW GRAPHS

PROCESS OF BUILDING THE TOOL:

- Learnt about ***Kturtle*** language from this link, about syntax for loops and assignment statements and different keywords.
- Made several simple small turtle code snippets and observed the resulting IR structure and understood the format and significance/meaning of each part.
- Learnt about ***graphviz*** library , digraph and its attribute and how to build ***dot*** file and render it was understood from various online resources.
- Using the '***relative sequence number***' type of statement was determined and after determining the type using '***relative sequence number***' and '***sequence number***' analysis of the generated ***Intermediate Representation*** was done.
- Based on the analysis, ***Control Flow Graph*** for the turtle file was created and outputted to user in two formats, that is ***dot*** file and ***png*** file (visual representation)

EXECUTION FLOW OF THE TOOL:

- User inputs the turtle file path as command line argument to the ***Kachua.py***, file is then opened and parsed to make the ***Intermediate Representation (IR).***
- ***IR*** is passed to the ***Submission.py*** function ***genCFG()*** by ***Kachua.py*** to build a ***CFG***.
- First ***targetcalc()*** is called to determine target statements for conditional and unconditional jump statements of the input program
- A for loop reads each statement in the ***IR*** and based on the ***relative sequence number***, statements are grouped into ***nodes (using genNode())*** and appropriate ***edges*** are decided.
- After determining all nodes and edges, they are passed as list to ***Kachua.py*** , imitating a graph like data structure.
- ***Kachua.py*** then passes the file name and ***CFG*** data structure obtained from ***genCFG()*** to ***dumpCFG()***
- ***dumpCFG()*** takes the ***CFG*** data structure and builds a ***dot*** file and renders a ***Control Flow Graph*** and stores it as a ***png*** file.

STEPS TO RUN THE TOOL:

1. Check if python packages like ***graphviz, kturtle*** and other packages and dependencies mentioned in ***Readme.md*** are installed properly, if not you can install it by running command "***pip <space> install <space> <package name>***" in any terminal or appropriate commands from ***Readme.md***
2. Open any terminal (***Command Prompt, Windows PowerShell, Git Bash***, etc.) and move into the directory where ***Kachua.py*** is present
3. Type " ***./kachua.py <space> file path/turtle file name.tl***"
   For eg, to run one of the testcases provided type: ***./kachua.py <space> ../Submission/testcases/mytest.tl***
4. Press Enter
5. ***IR*** and ***dot*** file will be displayed in ***terminal*** and ***another window*** will contain ***Control Flow Graph visual representation.***
6. ***Dot*** file and ***png*** image are stored in ***Submission/testcases*** folder
   For eg, output of ***mytest*** is saved as ***cfg.dot*** and ***cfg.dot.png***

HARSH AGARWAL
21111030
MTECH CSE 2021-23
PROGRAM ANALYSIS, VERIFICATION AND TESTING