We made the fact that the domain is code

We decided to partition the document into code-segment, which constitute of:

1. F - Function implementation.
2. S - Struct definition.
3. V - Global variables, include and define statements (per file).

The underlying algorithm relies on two graphs, one for modeling the human interaction with the code, and the other for representing the code structure itself:

1. MIP-Net (**M**utual **I**nfluence **P**otential)
2. CSR-Net (**C**ode **S**egment’s **R**elations)

MIP-Net:

The concept of MIP-Net is explained in detail in Dr. Amir research paper, which is the basis to this project, as mentioned before.  
In this project, we used a (slightly modified) version of the graph,

The idea is, that the more a person interacted with a code segment, the important it is to them.

CSR-Net:

We defined this graph to model the interaction between object’s (code-segment’s) stem from the structured grammar of a programing language.

We denote a set of relations between codes-segments as a collection of pairs where is a condition on the connected nodes, and is a numerical value.

The nodes are code-segment’s (in the code: identified by name).

The edges are weighted according to the relationship between the nodes.  
Formally, .

The relation’s in :

triplet: () where are types of object () that the weight (w) will be added

The weight of a given edge will be the sum of all the matching weights out of the possible triplets are as follow: