CSE 6708 - Semantic Web

Technical Documentation on Assignment 1

Paper Name: Source Code Plagiarism Detection Method Using Protégé Built Ontologies

Samidhya Sarker
Student No. 1018052049
Group-2

September 6, 2019

Contents

List of Figures

Listings

1

Introduction

Software Plagiarism is defined as Copying Software without giving attribution. Ion Smeureanu and Bogdan Iancu of the The Bucharest University of Economic Studies have written a scientific paper. In this paper, the authors devised a use of semantic web technologies to prevent software

1.1 Goals

1. Matching authors software implementation:
    - Create ontologies from source code manually by hand using protege.
    - Execute sparql queries on ontologies and compare the metrics.
    - Create topic maps using Protege OntoGraf plugin.
2. Doing further works as dictated by the authors.
    - Create a parser.

2

Emulating Experiments of Paper Authors

2.1 Creating ontologies

2.1.1 Tools Used

- Protege 5.5.0 with OWL Code Generation Plug-in (2.0.0)
- VIM 8.1 with niklasl/vim-rdf and n3.vim Plug-in

2.1.2 Source Code

The authors of the paper had given two source codes in the paper that is identical in resulting output but different lexically. The source code takes 3 numbers as input and outputs the maximum.

One is in the C programming language.

```c
#include <stdio.h>

int option = 0;
int i;
int numbers[3];

int main (){
    while (option!=3){
        printf("Please choose an option and press enter:\n");
        printf("1. Read 3 numbers\n 2. Print the max\n 3.Exit\n
    ↪ ");
        scanf("%i",&option);
        if (option==1) {
            for (i=0; i<3; i++) {
                printf("\nnumbers[%i]=",i+1);
                scanf("%i",&numbers[i]);
            }
        } else if (option==2) {
            int max = 0;
            for (i=0; i<3; i++) {
                if(numbers[i] > max) {
                    max = numbers[i];
                }
            }
            printf("\nMax=%i",max);
        }
    }
}
```

Listing 2.1: C source code for the max out of 3 program

And another is in the Javascript programming language.

```
var option = 0;
var i = 0;
var numbers = new Array();

while (option!=3){
    document.write("Please choose an option and press enter:\n"
    ↪ );
    document.write("1. Read 3 numbers\n 2. Print the max\n 3.
    ↪ Exit\n");
    option = prompt("Option");
    if (option == 1) {
        for (i=0; i<3; i++) {
            numbers[i] = prompt("numbers[" + (i+1) + "]");
        }
    } else if (option == 2) {
        var max = 0;
        for (i=0; i<3; i++) {
            if(numbers[i] > max) {
                max = numbers[i];
            }
        }
        document.write("\nMax=" + max + "\n");
    }
}
```

Listing 2.2: JavaScript source code for the max out of 3 program

2.1.3 Ontologies

We convert the source code into schema ontologies by defining Classes, object properties, data properties and individual code elements. The basic schema was given by authors and we extended it.

We used RDF/XML OWL format for representing source ontologies because Protege is better suited for this than RDF n3. it We get ontology 4.1for the source code defined in 2.1.

We also got a OWL ontology 4.2 created from our javascript source code defined in 2.2

2.2 SPARQL Query

2.2.1 Tools Used

- Protege 5.5.0 with SPARQL Query Plug-in (3.0.0)
- VIM 8.1 with omer/vim-sparql Plug-In

We used SPARQL Plug-In for Protege for executing sparql query in ontology 4.1 and 4.2. Instead we could have used Apache Jena Fuseki server [1] for creating a SPARQL API endpoint.

We ran the following 8 SPARQL queries separately and got the expected results.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

# For C source code
BASE <https://semantic-web.netlify.com/c_source_ontology/>

# For Javascript Code
BASE <https://semantic-web.netlify.com/js_source_ontology/>


# 1. Total number of conditional structures

SELECT (COUNT(?subject) AS ?c) WHERE {
    ?subject rdf:type <ConditionalStructure>
}

# Expected Result on C code: 3
# Expected Result on JS code: 3

# 2. Total number of repetitive structures

SELECT (COUNT(?subject) AS ?c) WHERE {
    ?subject rdf:type <RepetitiveStructure>
}

# Expected Result on C code: 3
# Expected Result on JS code: 3

# 3. Total number of variables

SELECT (COUNT(?subject) AS ?c) WHERE {
    ?subject rdf:type <Variable>
}


```

[1]https://jena.apache.org/

```
# Expected Result on C code: 4
# Expected Result on JS code: 4

# 4. Total number of conditional structures included in repetitive structures


SELECT (COUNT(?subject) AS ?c) WHERE {
    ?object rdf:type <RepetitiveStructure> .
    ?subject <is_included_in> ?object .
    ?subject rdf:type <ConditionalStructure>
}

# Expected Result on C code: 3
# Expected Result on JS code: 3

# 5. Total number of repetitive structures included in repetitive structures

SELECT (COUNT(?subject) AS ?c) WHERE {
    ?object rdf:type <RepetitiveStructure> .
    ?subject <is_included_in> ?object .
    ?subject rdf:type <RepetitiveStructure>
}

# Expected Result on C code: 0
# Expected Result on JS code: 0

# 6.  Total number of system functions called

SELECT (COUNT(?subject) AS ?c) WHERE {
    ?subject rdf:type <SystemFunction>
}

# Expected Result on C code: 5
# Expected Result on JS code: 4

# 7.  Total number of system functions called in conditional structures

SELECT (COUNT(?subject) AS ?c) WHERE {
    ?object rdf:type <ConditionalStructure> .
    ?subject <is_included_in> ?object .
    ?subject rdf:type <SystemFunction>
 }

 # Expected Result on C code: 1
 # Expected Result on JS code: 1

# 8. Total Number of repetative structures

SELECT (COUNT(?subject) AS ?c) WHERE {
    ?object rdf:type <RepetitiveStructure> .
    ?subject <is_included_in> ?object .
    ?subject rdf:type <SystemFunction>
}

# Expected Result on C code: 4
# Expected Result on JS code: 3
```

Listing 2.3: SPARQL Queries given by the paper authors

We matched the results given by the authors concluded the experiment as a success.

2.3 Topic Maps

2.3.1 Tools Used

- Protege 5.5.0 with OntoGraf Plug-in (2.0.3)

For creating topic maps of the object properties of found ontologies, we used the OntoGraf Plug-In. We only showed the individual nodes not the class nodes and the properties associated with them as edges between nodes. We got the folling map of the C source ontology at 4.1 found from source 2.1.

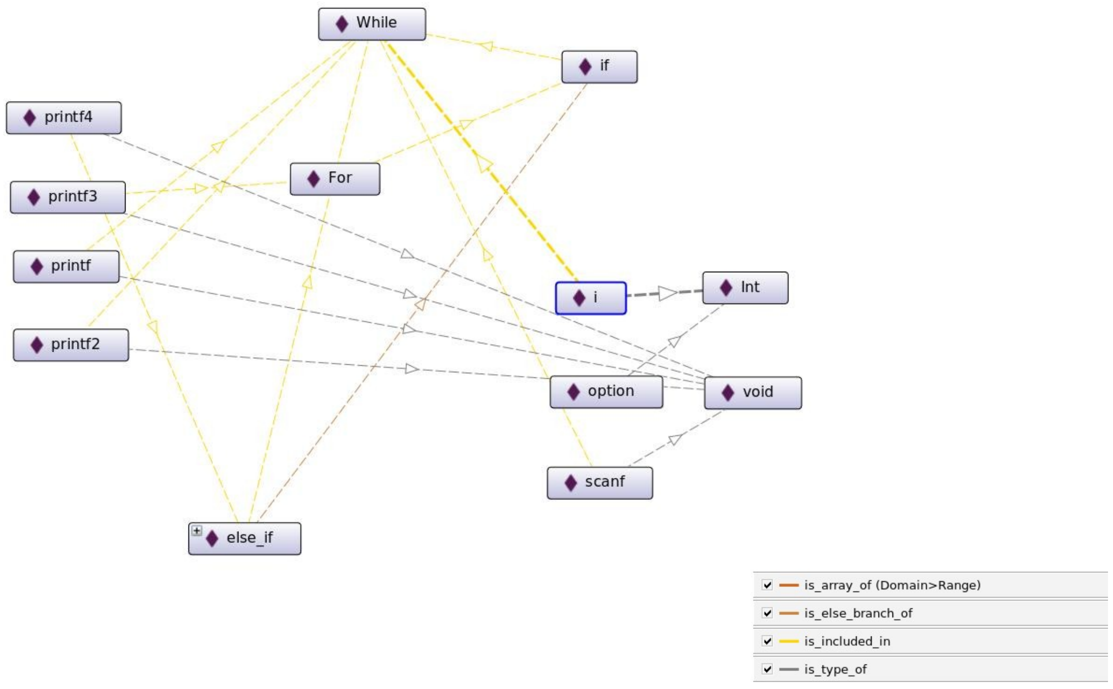

Figure 2.1: Topic map ontology created by source code written in 2.1 in C language

In the same way we got the topic maps for the source 2.2 for javascript language.

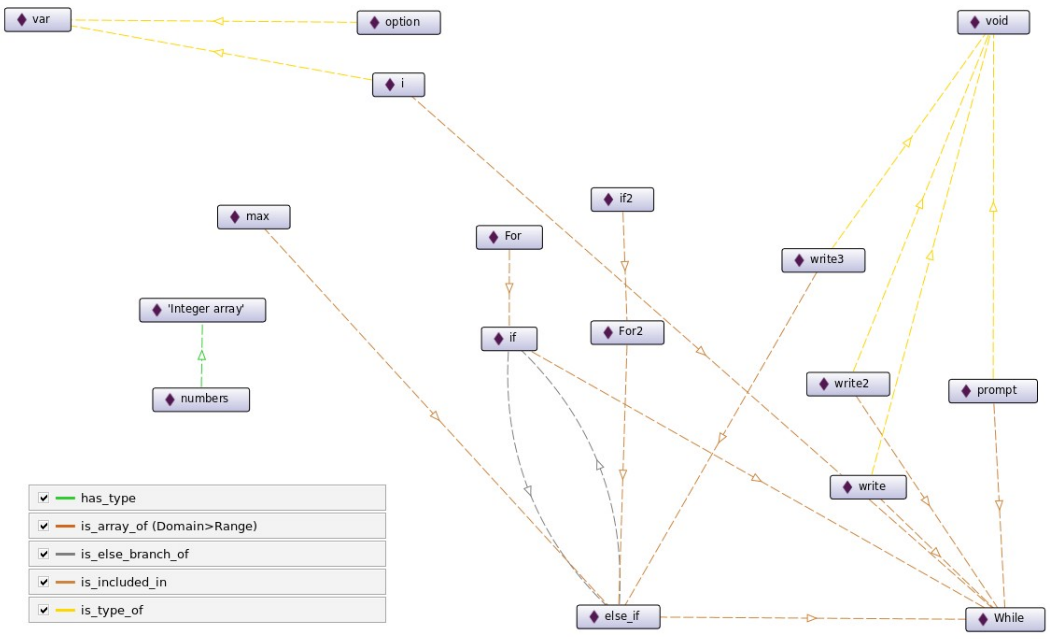

Figure 2.2: Topic map ontology created by source code written in 2.2 in JS

2.4 Conclusion

Comparing the SPARQL query results and topic maps we can decide that the source code of 2.1 and 2.2 has been plagiarized one from another.

3

Implementation of specified further works of the paper

In the further works section of the stated paper, the authors highlighted the following works that could be done:

1. A parser for code.
2. A crawler for parsing code.
3. Dynamically generated SPARQL query for defined metrics.

The creation of parser is the first step for the creation of an automated plagiarism detection system as envisioned by the authors. So, my research was focused on the formulation of a parser generator.

3.1 An existing RDFized parser generator for the JAVA programming language: Codeontology

The paper assigned to group-1 [1] [2] already provides a RDF triple generating parser [3]. So for testing if it can cater to our needs, I transliterated out source 2.1 and 2.2 into Java programming language listed 3.1

```java
import java.util.Scanner;
public class MaxOfThreeNumbers {
    public static void main(String[] args) {
        int option = 0;
        int i;
        int[] numbers = new int[3];

        while (option!=3){
            System.out.println("Please choose an option and
    ↪ press enter:\n");
            System.out.println("1. Read 3 numbers\n 2. Print
    ↪ the max\n 3.Exit\n");

            Scanner scan = new Scanner(System.in);
            option = scan.nextInt();

            if (option==1) {
                for (i=0; i<3; i++) {
                    System.out.printf("\nnumbers[%d]=",i+1);
                    numbers[i] = scan.nextInt();
                }
            } else if (option==2) {
                int max = 0;
                for (i=0; i<3; i++) {
                    if(numbers[i] > max) {
                        max = numbers[i];
                    }
                }
                System.out.printf("\nMax=%d",max);
            }
        }


    }
}
```

Listing 3.1: Java source code for the max out of 3 program

The main goal of the Codeontology project is to generate knowledge base for complex Java projects that is different from our goal of detecting minute information of source codes. It does not take account of programming structures, only local variables, system functions and informations related to Java

[1] CodeOntology: RDF-ization of Source Code
[2] `http://codeontology.org/`
[3] `https://github.com/codeontology/parser`

programming environments like Package and Streams etc. This project can be modified to our liking and can also be adapted to other programming languages like C, Javascript.

We get a highly complex topic map different that 2.3.1 and 2.3.1.

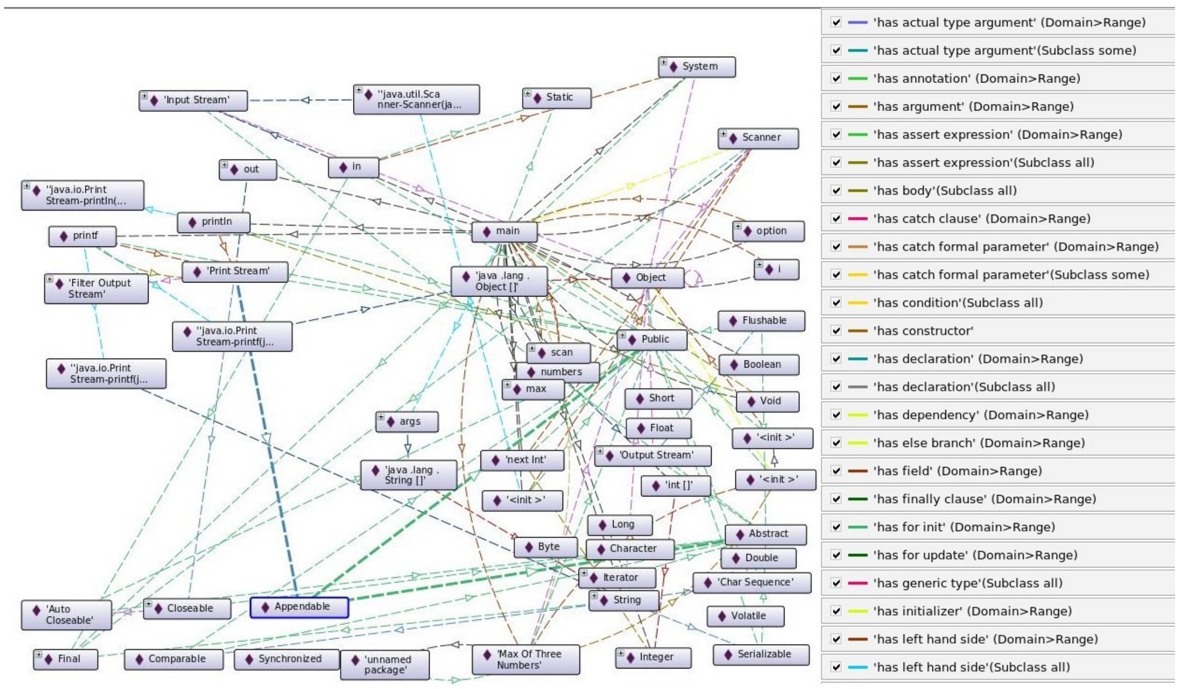
'has actual type argument' (Domain>Range)
'has actual type argument'(Subclass some)
'has annotation' (Domain>Range)
'has argument' (Domain>Range)
'has assert expression' (Domain>Range)
'has assert expression'(Subclass all)
'has body'(Subclass all)
'has catch clause' (Domain>Range)
'has catch formal parameter' (Domain>Range)
'has catch formal parameter'(Subclass some)
'has condition'(Subclass all)
'has constructor'
'has declaration' (Domain>Range)
'has declaration'(Subclass all)
'has dependency' (Domain>Range)
'has else branch' (Domain>Range)
'has field' (Domain>Range)
'has finally clause' (Domain>Range)
'has for init' (Domain>Range)
'has for update' (Domain>Range)
'has generic type'(Subclass all)
'has initializer' (Domain>Range)
'has left hand side' (Domain>Range)
'has left hand side'(Subclass all)
System
'Input Stream'
''java.util.Scanner-Scanner(ja...
Static
Scanner
out
in
''java.io.Print Stream-println(...
println
main
option
printf
'Print Stream'
'java .lang . Object []'
Object
i
'Filter Output Stream'
''java.io.Print Stream-printf(j...
Flushable
Public
''java.io.Print Stream-printf(j...
scan
Boolean
numbers
max
Short
Void
args
Float
'<init >'
'next Int'
'Output Stream'
'java .lang . String []'
'int []'
'<init >'
'<init >'
Long
Abstract
Byte
Character
Double
Iterator
'Char Sequence'
'Auto Closeable'
Closeable
Appendable
String
Volatile
Final
Comparable
Synchronized
'unnamed package'
'Max Of Three Numbers'
Integer
Serializable

Figure 3.1: Topic map ontology created by source code written in 3.1 in JAVA programming language

3.2 Creating a Parser generator using ANTLR

ANTLR is a language recognition toolset which uses LL(*) grammer for parsing. There are language grammers available including C [4], JavaScript [5], Java [6] etc. We used antlr to create a parse tree for the C soruce described in 2.1

Also, a parse tree for Javascript source from 2.2 was also created.

So, ANTLR can certainly recognize C and JS source files. By modifying antler listener classes we can create toolset to generate dynamic rdf/xml files.

[4]https://github.com/antlr/grammars-v4/tree/master/c

[5]https://github.com/antlr/grammars-v4/tree/master/ecmascript

[6]https://github.com/antlr/grammars-v4/tree/master/java

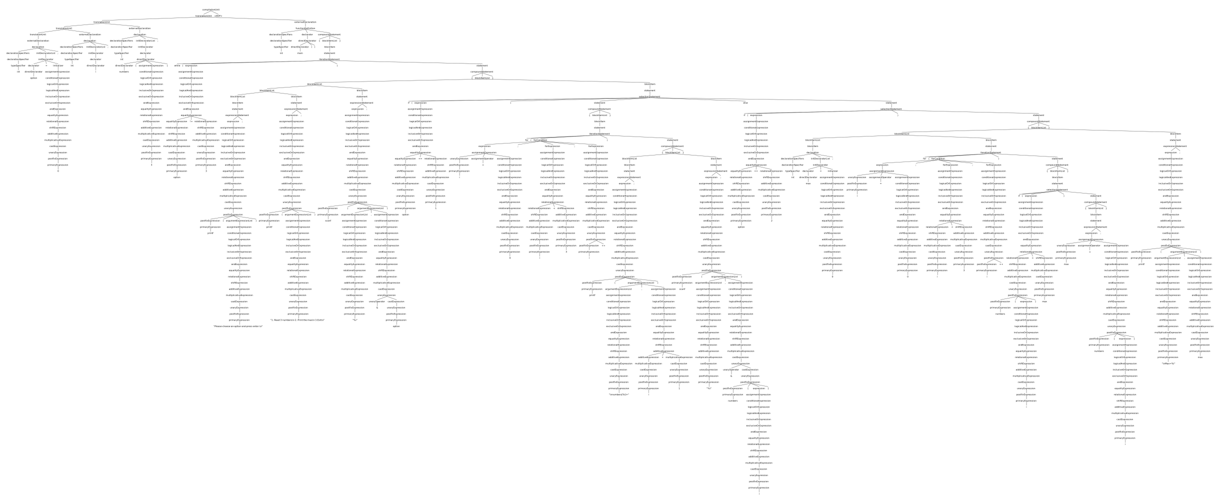
Figure 3.2: ANTLR parse tree created from C source at 2.1

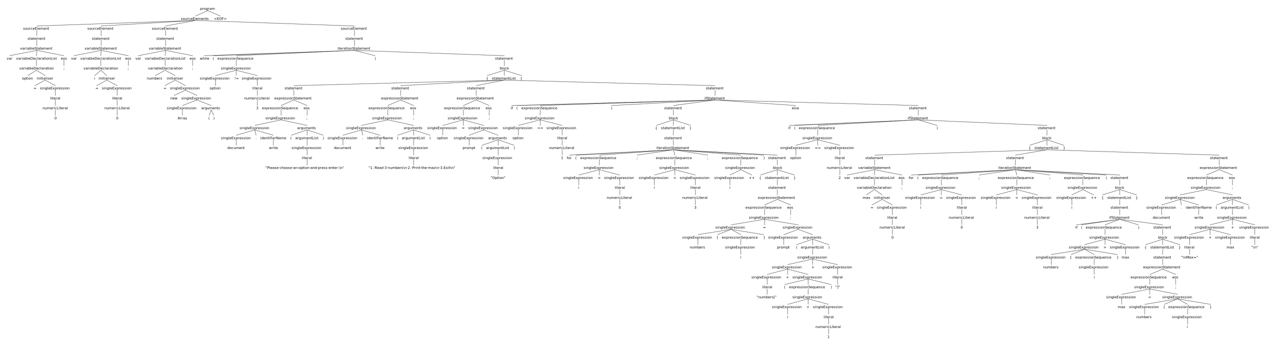
Figure 3.3: ANTLR parse tree created from JS source at 2.2

3.3 Modifying Flex/Bison Code Generators to create RDF/XML generators

Hypothetically, one can use the code generation techniques learned in CSE-310 course and modify the Assignment-4 (Code Generation) so that the parser generator outputs RDF/XML instead of ASSEMBLY code.
[7]

[7] 3.2 and 3.3 has not been fully implemented due to time shortage.

4

Appendix

4.1 Owl soruce code for C source code defined in 2.1

```
<?xml version="1.0"?>
<rdf:RDF xmlns="https://semantic-web.netlify.com/
   ↪ c_source_ontology/"
     xml:base="https://semantic-web.netlify.com/
   ↪ c_source_ontology/"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:xml="http://www.w3.org/XML/1998/namespace"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:xkos="http://rdf-vocabulary.ddialliance.org/xkos#"
     xmlns:dcterms="http://purl.org/dc/terms/"
     xmlns:c_source_ontology="https://semantic-web.netlify.com/
   ↪ c_source_ontology/">
    <owl:Ontology rdf:about="https://semantic-web.netlify.com/
   ↪ c_source_ontology/">
        <rdfs:comment xml:lang="en">An ontology that represents
   ↪  structural and imperative programming languages</
   ↪ rdfs:comment>
        <rdfs:label xml:lang="en">SoruceCodeOntology</
   ↪ rdfs:label>
    </owl:Ontology>



    <!--

   ↪ ///////////////////////////////////////////////////////////////////////////////////
   ↪
    //
    // Object Properties
    //

```

```
↪ ////////////////////////////////////////////////////////////////////////////////
↪
  -->




<!-- https://semantic-web.netlify.com/c_source_ontology/
↪ conditions -->

<owl:ObjectProperty rdf:about="https://semantic-web.netlify
↪ .com/c_source_ontology/conditions">
    <owl:inverseOf rdf:resource="https://semantic-web.
↪ netlify.com/c_source_ontology/has_condition"/>
    <rdfs:domain>
        <owl:Restriction>
            <owl:onProperty rdf:resource="https://semantic-
↪ web.netlify.com/c_source_ontology/conditions"/>
            <owl:someValuesFrom rdf:resource="https://
↪ semantic-web.netlify.com/c_source_ontology/
↪ ConditionalStructure"/>
        </owl:Restriction>
    </rdfs:domain>
    <rdfs:range>
        <owl:Restriction>
            <owl:onProperty rdf:resource="https://semantic-
↪ web.netlify.com/c_source_ontology/conditions"/>
            <owl:someValuesFrom rdf:resource="https://
↪ semantic-web.netlify.com/c_source_ontology/
↪ RepetitiveStructure"/>
        </owl:Restriction>
    </rdfs:range>
</owl:ObjectProperty>



<!-- https://semantic-web.netlify.com/c_source_ontology/
↪ has_condition -->

<owl:ObjectProperty rdf:about="https://semantic-web.netlify
↪ .com/c_source_ontology/has_condition"/>



<!-- https://semantic-web.netlify.com/c_source_ontology/
↪ has_type -->

<owl:ObjectProperty rdf:about="https://semantic-web.netlify
↪ .com/c_source_ontology/has_type">
    <owl:inverseOf rdf:resource="https://semantic-web.
↪ netlify.com/c_source_ontology/is_type_of"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#
↪ FunctionalProperty"/>
    <rdfs:range rdf:resource="https://semantic-web.netlify.
```

```
com/c_source_ontology/DataType"/>
    </owl:ObjectProperty>



    <!-- https://semantic-web.netlify.com/c_source_ontology/is_array_of -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/c_source_ontology/is_array_of">
        <rdfs:domain rdf:resource="https://semantic-web.netlify.com/c_source_ontology/Variable"/>
        <rdfs:range rdf:resource="https://semantic-web.netlify.com/c_source_ontology/DataType"/>
    </owl:ObjectProperty>



    <!-- https://semantic-web.netlify.com/c_source_ontology/is_else_branch_of -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/c_source_ontology/is_else_branch_of">
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#InverseFunctionalProperty"/>
        <rdfs:comment>I have confusion about the Characteristics of the function.</rdfs:comment>
    </owl:ObjectProperty>



    <!-- https://semantic-web.netlify.com/c_source_ontology/is_included_in -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/c_source_ontology/is_included_in"/>



    <!-- https://semantic-web.netlify.com/c_source_ontology/is_type_of -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/c_source_ontology/is_type_of"/>



    <!--
    /////////////////////////////////////////////////////////////////////////////
    //
    // Data properties
    //
   ↪ /////////////////////////////////////////////////////////////////////////////
   ↪
     -->




    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ dimension -->

    <owl:DatatypeProperty rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/dimension">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org
   ↪ /2002/07/owl#topDataProperty"/>
    </owl:DatatypeProperty>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ name -->

    <owl:DatatypeProperty rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/name">
        <rdfs:domain>
            <owl:Class>
                <owl:unionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="https://
   ↪ semantic-web.netlify.com/c_source_ontology/Constants"/>
                    <rdf:Description rdf:about="https://
   ↪ semantic-web.netlify.com/c_source_ontology/DataType"/>
                </owl:unionOf>
            </owl:Class>
        </rdfs:domain>
        <rdfs:range rdf:resource="http://www.w3.org/2001/
   ↪ XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ source_code -->

    <owl:DatatypeProperty rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/source_code">
        <rdfs:domain>
            <owl:Class>
                <owl:unionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="https://
   ↪ semantic-web.netlify.com/c_source_ontology/
   ↪ ConditionalStructure"/>
                    <rdf:Description rdf:about="https://
```

```
↪ semantic-web.netlify.com/c_source_ontology/Constants"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/c_source_ontology/DataType"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/c_source_ontology/Operator"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/c_source_ontology/
↪ ProgrammingStructure"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/c_source_ontology/
↪ RepetitiveStructure"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/c_source_ontology/SystemFunction
↪ "/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/c_source_ontology/Variable"/>
                </owl:unionOf>
            </owl:Class>
        </rdfs:domain>
        <rdfs:range rdf:resource="http://www.w3.org/2001/
↪ XMLSchema#string"/>
        <rdfs:label xml:lang="en">source code</rdfs:label>
    </owl:DatatypeProperty>



    <!--

↪ ///////////////////////////////////////////////////////////////////////////////
↪
    //
    // Classes
    //

↪ ///////////////////////////////////////////////////////////////////////////////
↪
     -->




    <!-- https://semantic-web.netlify.com/c_source_ontology/
↪ Array -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ c_source_ontology/Array">
        <rdfs:subClassOf rdf:resource="https://semantic-web.
↪ netlify.com/c_source_ontology/DataType"/>
    </owl:Class>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
↪ Comment -->
```

```xml

    <owl:Class rdf:about="https://semantic-web.netlify.com/c_source_ontology/Comment"/>



    <!-- https://semantic-web.netlify.com/c_source_ontology/ConditionalStructure -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/c_source_ontology/ConditionalStructure">
        <rdfs:subClassOf rdf:resource="https://semantic-web.netlify.com/c_source_ontology/ProgrammingStructure"/>
    </owl:Class>



    <!-- https://semantic-web.netlify.com/c_source_ontology/Constants -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/c_source_ontology/Constants"/>



    <!-- https://semantic-web.netlify.com/c_source_ontology/DataType -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/c_source_ontology/DataType"/>



    <!-- https://semantic-web.netlify.com/c_source_ontology/Operator -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/c_source_ontology/Operator"/>



    <!-- https://semantic-web.netlify.com/c_source_ontology/ProgrammingStructure -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/c_source_ontology/ProgrammingStructure"/>



    <!-- https://semantic-web.netlify.com/c_source_ontology/RepetitiveStructure -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/c_source_ontology/RepetitiveStructure">
```

```
        <rdfs:subClassOf rdf:resource="https://semantic-web.
    ↪ netlify.com/c_source_ontology/ProgrammingStructure"/>
    </owl:Class>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
    ↪ SystemFunction -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
    ↪ c_source_ontology/SystemFunction"/>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
    ↪ Variable -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
    ↪ c_source_ontology/Variable"/>



    <!--

    ↪ ////////////////////////////////////////////////////////////////////////////////
    ↪
    //
    // Individuals
    //

    ↪ ////////////////////////////////////////////////////////////////////////////////
    ↪
     -->




    <!-- https://semantic-web.netlify.com/c_source_ontology/For
    ↪  -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/c_source_ontology/For">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/c_source_ontology/RepetitiveStructure"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/c_source_ontology/if"/>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">for (i=0; i&lt;3; i++) {
        printf(&quot;\nnumbers[%i]=&quot;,i+1);scanf(&quot;%i&
    ↪ quot;,&amp;numbers[i]);
}</source_code>
    </owl:NamedIndividual>


```

```xml

    <!-- https://semantic-web.netlify.com/c_source_ontology/For2 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/For2">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/RepetitiveStructure"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/c_source_ontology/else_if"/>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">for (i=0; i&lt;3; i++) {
     if(numbers[i] &gt; max) {
             max = numbers[i];
     }
}</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/Int -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/Int">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/DataType"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">int</name>
        <rdfs:comment xml:lang="en">The int type</rdfs:comment>
        <rdfs:label xml:lang="en">Int</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/Void -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/Void">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/DataType"/>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">void</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/While -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/While">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/c_source_ontology/RepetitiveStructure"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
    ↪ string">while</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">while (option!=3){
    printf(&quot;Please choose an option and press enter:\n&
    ↪ quot;);
    printf(&quot;1. Read 3 numbers\n 2. Print the max\n 3.Exit\
    ↪ n&quot;);
    scanf(&quot;%i&quot;,&amp;option);
    if (option==1) {
        for (i=0; i&lt;3; i++) {
            printf(&quot;\nnumbers[%i]=&quot;,i+1);scanf(&quot
    ↪ ;%i&quot;,&amp;numbers[i]);
        }
    } else if (option==2) {
        int max = 0;
        for (i=0; i&lt;3; i++) {
            if(numbers[i] &gt; max) {
                max = numbers[i];
            }
        }
        printf(&quot;\nMax=%i&quot;,max);
    }
}</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
    ↪ else_if -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/c_source_ontology/else_if">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/c_source_ontology/ConditionalStructure"/>
        <is_else_branch_of rdf:resource="https://semantic-web.
    ↪ netlify.com/c_source_ontology/if"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/c_source_ontology/While"/>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">else if (option==2) {
        int max = 0;
        for (i=0; i&lt;3; i++) {
            if(numbers[i] &gt; max) {
                max = numbers[i];
            }
        }
        printf(&quot;\nMax=%i&quot;,max);
    }</source_code>
    </owl:NamedIndividual>



```

```xml
    <!-- https://semantic-web.netlify.com/c_source_ontology/i -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/i">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/Variable"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/c_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.com/c_source_ontology/Int"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">i</name>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">int i</source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">i</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/if -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/if">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/ConditionalStructure"/>
        <is_else_branch_of rdf:resource="https://semantic-web.netlify.com/c_source_ontology/else_if"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/c_source_ontology/While"/>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">if (option==1) {
        for (i=0; i&lt;3; i++) {
            printf(&quot;\nnumbers[%i]=&quot;,i+1);scanf(&quot;%i&quot;,&amp;numbers[i]);
        }
    }</source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">if</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/if2 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/if2">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/ConditionalStructure"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/c_source_ontology/For2"/>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">if(numbers[i] &gt; max) {
    max = numbers[i];
}</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/int_array -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/int_array">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/Array"/>
        <dimension rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</dimension>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">integer array</name>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Integer array</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/max -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/max">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/Variable"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/c_source_ontology/else_if"/>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/numbers -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/c_source_ontology/numbers">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/Variable"/>
        <has_type rdf:resource="https://semantic-web.netlify.com/c_source_ontology/int_array"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">numbers</name>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">int numbers[3]</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ option -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/option">
        <rdf:type rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/Variable"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/Int"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
   ↪ string">option</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
   ↪ XMLSchema#string">int option = 0</source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/
   ↪ XMLSchema#string">option</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ printf -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/printf">
        <rdf:type rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
   ↪ netlify.com/c_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
   ↪ string">printf</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
   ↪ XMLSchema#string">printf(&quot;Please choose an option
   ↪ and press enter:\n&quot;);</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ printf2 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/printf2">
        <rdf:type rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
   ↪ netlify.com/c_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
   ↪ string">printf</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
   ↪ XMLSchema#string">printf(&quot;1. Read 3 numbers\n 2.
   ↪ Print the max\n 3.Exit\n&quot;);</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ printf3 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/printf3">
        <rdf:type rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
   ↪ netlify.com/c_source_ontology/For"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
   ↪ string">printf</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
   ↪ XMLSchema#string">printf(&quot;\nnumbers[%i]=&quot;,i+1);
   ↪ scanf(&quot;%i&quot;,&amp;numbers[i]);</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ printf4 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/printf4">
        <rdf:type rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
   ↪ netlify.com/c_source_ontology/else_if"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
   ↪ string">printf</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
   ↪ XMLSchema#string">printf(&quot;\nMax=%i&quot;,max);</
   ↪ source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/c_source_ontology/
   ↪ scanf -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
   ↪ netlify.com/c_source_ontology/scanf">
        <rdf:type rdf:resource="https://semantic-web.netlify.
   ↪ com/c_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
```

```
netlify.com/c_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.com/c_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">scanf</name>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">scanf(&quot;%i&quot;,&amp;option);</source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">scanf</rdfs:label>
    </owl:NamedIndividual>
</rdf:RDF>



<!-- Generated by the OWL API (version 4.5.9.2019-02-01T07:24:44Z) https://github.com/owlcs/owlapi -->
```

Listing 4.1: Owl source code for C source code defined in 2.1

4.2 Owl soruce code for JavaScript source code defined in 2.2

```
<?xml version="1.0"?>
<rdf:RDF xmlns="https://semantic-web.netlify.com/js_source_ontology/"
     xml:base="https://semantic-web.netlify.com/js_source_ontology/"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:xml="http://www.w3.org/XML/1998/namespace"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:xkos="http://rdf-vocabulary.ddialliance.org/xkos#"
     xmlns:dcterms="http://purl.org/dc/terms/"
     xmlns:c_source_ontology="https://semantic-web.netlify.com/c_source_ontology/">
    <owl:Ontology rdf:about="https://semantic-web.netlify.com/js_source_ontology/">
        <rdfs:comment xml:lang="en">An ontology that represents structural and imperative programming languages</rdfs:comment>
        <rdfs:label xml:lang="en">CodeOntology_Modified</rdfs:label>
    </owl:Ontology>



    <!--

    ///////////////////////////////////////////////////////////////////////////////
    //
    // Object Properties
    //
    ////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- https://semantic-web.netlify.com/js_source_ontology/conditions -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/conditions">
        <owl:inverseOf rdf:resource="https://semantic-web.netlify.com/js_source_ontology/has_condition"/>
        <rdfs:domain>
            <owl:Restriction>
                <owl:onProperty rdf:resource="https://semantic-web.netlify.com/js_source_ontology/conditions"/>
                <owl:someValuesFrom rdf:resource="https://semantic-web.netlify.com/js_source_ontology/ConditionalStructure"/>
            </owl:Restriction>
        </rdfs:domain>
        <rdfs:range>
            <owl:Restriction>
                <owl:onProperty rdf:resource="https://semantic-web.netlify.com/js_source_ontology/conditions"/>
                <owl:someValuesFrom rdf:resource="https://semantic-web.netlify.com/js_source_ontology/RepetitiveStructure"/>
            </owl:Restriction>
        </rdfs:range>
    </owl:ObjectProperty>



    <!-- https://semantic-web.netlify.com/js_source_ontology/has_condition -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/has_condition"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/has_type -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/has_type">
        <owl:inverseOf rdf:resource="https://semantic-web.netlify.com/js_source_ontology/is_type_of"/>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
        <rdfs:range rdf:resource="https://semantic-web.netlify.com/js_source_ontology/DataType"/>
    </owl:ObjectProperty>



    <!-- https://semantic-web.netlify.com/js_source_ontology/is_array_of -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/is_array_of">
        <rdfs:domain rdf:resource="https://semantic-web.netlify.com/js_source_ontology/Variable"/>
        <rdfs:range rdf:resource="https://semantic-web.netlify.com/js_source_ontology/DataType"/>
    </owl:ObjectProperty>



    <!-- https://semantic-web.netlify.com/js_source_ontology/is_else_branch_of -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/is_else_branch_of">
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#InverseFunctionalProperty"/>
        <rdfs:comment>I have confusion about the Characteristics of the function.</rdfs:comment>
    </owl:ObjectProperty>



    <!-- https://semantic-web.netlify.com/js_source_ontology/is_included_in -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/is_included_in"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/is_type_of -->

    <owl:ObjectProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/is_type_of"/>



    <!--

```

```
↪ /////////////////////////////////////////////////////////////////////////////
↪
    //
    // Data properties
    //
    /////////////////////////////////////////////////////////////////////////////
     -->




    <!-- https://semantic-web.netlify.com/js_source_ontology/dimension -->

    <owl:DatatypeProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/dimension">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"/>
    </owl:DatatypeProperty>



    <!-- https://semantic-web.netlify.com/js_source_ontology/name -->

    <owl:DatatypeProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/name">
        <rdfs:domain>
            <owl:Class>
                <owl:unionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="https://semantic-web.netlify.com/js_source_ontology/Constants"/>
                    <rdf:Description rdf:about="https://semantic-web.netlify.com/js_source_ontology/DataType"/>
                </owl:unionOf>
            </owl:Class>
        </rdfs:domain>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- https://semantic-web.netlify.com/js_source_ontology/source_code -->

    <owl:DatatypeProperty rdf:about="https://semantic-web.netlify.com/js_source_ontology/source_code">
        <rdfs:domain>
            <owl:Class>
                <owl:unionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="https://
```

```
↪ semantic-web.netlify.com/js_source_ontology/
↪ ConditionalStructure"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/js_source_ontology/Constants"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/js_source_ontology/DataType"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/js_source_ontology/Operator"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/js_source_ontology/
↪ ProgrammingStructure"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/js_source_ontology/
↪ RepetitiveStructure"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/js_source_ontology/
↪ SystemFunction"/>
                    <rdf:Description rdf:about="https://
↪ semantic-web.netlify.com/js_source_ontology/Variable"/>
                </owl:unionOf>
            </owl:Class>
        </rdfs:domain>
        <rdfs:range rdf:resource="http://www.w3.org/2001/
↪ XMLSchema#string"/>
        <rdfs:label xml:lang="en">source code</rdfs:label>
    </owl:DatatypeProperty>



    <!--
↪ //////////////////////////////////////////////////////////////////////////////
↪
    //
    // Classes
    //
↪ //////////////////////////////////////////////////////////////////////////////
↪
     -->




    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ Array -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ js_source_ontology/Array">
        <rdfs:subClassOf rdf:resource="https://semantic-web.
↪ netlify.com/js_source_ontology/DataType"/>
    </owl:Class>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ Comment -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ js_source_ontology/Comment"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ ConditionalStructure -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ js_source_ontology/ConditionalStructure">
        <rdfs:subClassOf rdf:resource="https://semantic-web.
↪ netlify.com/js_source_ontology/ProgrammingStructure"/>
    </owl:Class>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ Constants -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ js_source_ontology/Constants"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ DataType -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ js_source_ontology/DataType"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ Operator -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ js_source_ontology/Operator"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ ProgrammingStructure -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/
↪ js_source_ontology/ProgrammingStructure"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ RepetitiveStructure -->
```

```

    <owl:Class rdf:about="https://semantic-web.netlify.com/js_source_ontology/RepetitiveStructure">
        <rdfs:subClassOf rdf:resource="https://semantic-web.netlify.com/js_source_ontology/ProgrammingStructure"/>
    </owl:Class>



    <!-- https://semantic-web.netlify.com/js_source_ontology/SystemFunction -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/js_source_ontology/SystemFunction"/>



    <!-- https://semantic-web.netlify.com/js_source_ontology/Variable -->

    <owl:Class rdf:about="https://semantic-web.netlify.com/js_source_ontology/Variable"/>



    <!--
    ///////////////////////////////////////////////////////////////////////////////
    //
    // Individuals
    //
    ///////////////////////////////////////////////////////////////////////////////
     -->




    <!-- https://semantic-web.netlify.com/js_source_ontology/For -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/js_source_ontology/For">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/js_source_ontology/RepetitiveStructure"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/js_source_ontology/if"/>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">for (i=0; i&lt;3; i++) {
        document.write(&quot;\nnumbers[%i]=&quot;,i+1);scanf(&quot;%i&quot;,&amp;numbers[i]);
}</source_code>
```

```
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/For2 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/js_source_ontology/For2">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/js_source_ontology/RepetitiveStructure"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/js_source_ontology/else_if"/>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">for (i=0; i&lt;3; i++) {
     if(numbers[i] &gt; max) {
             max = numbers[i];
     }
}</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/Void -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/js_source_ontology/Void">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/js_source_ontology/DataType"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">void</name>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">void</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/While -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/js_source_ontology/While">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/js_source_ontology/RepetitiveStructure"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">while</name>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">while (option!=3){
    document.write(&quot;Please choose an option and press enter:\n&quot;);
    document.write(&quot;1. Read 3 numbers\n 2. Print the max\n 3.Exit\n&quot;);
    option = prompt(&quot;Option&quot;);
    if (option == 1) {
        for (i=0; i&lt;3; i++) {
            numbers[i] = prompt(&quot;numbers[&quot; + (i+1) +
    ↪ &quot;]&quot;);
        }
    } else if (option == 2) {
        var max = 0;
        for (i=0; i&lt;3; i++) {
            if(numbers[i] &gt; max) {
                max = numbers[i];
            }
        }
        document.write(&quot;\nMax=&quot; + max + &quot;\n&quot
    ↪ ;);
    }
}</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ array -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/array">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/Array"/>
        <dimension rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#int">1</dimension>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
    ↪ string">array</name>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">Integer array</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ else_if -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/else_if">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/ConditionalStructure"/>
        <is_else_branch_of rdf:resource="https://semantic-web.
    ↪ netlify.com/js_source_ontology/if"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/js_source_ontology/While"/>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">else if (option==2) {
        var max = 0;
        for (i=0; i&lt;3; i++) {
            if(numbers[i] &gt; max) {
                max = numbers[i];
            }
        }
        document.write(&quot;\nMax=%i&quot;,max);
    }</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/i -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/js_source_ontology/i">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/js_source_ontology/Variable"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/js_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.com/js_source_ontology/var"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">i</name>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">int i</source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">i</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/if -->

    <owl:NamedIndividual rdf:about="https://semantic-web.netlify.com/js_source_ontology/if">
        <rdf:type rdf:resource="https://semantic-web.netlify.com/js_source_ontology/ConditionalStructure"/>
        <is_else_branch_of rdf:resource="https://semantic-web.netlify.com/js_source_ontology/else_if"/>
        <is_included_in rdf:resource="https://semantic-web.netlify.com/js_source_ontology/While"/>
        <source_code rdf:datatype="http://www.w3.org/2001/XMLSchema#string">if (option==1) {
        for (i=0; i&lt;3; i++) {
            document.write(&quot;\nnumbers[%i]=&quot;,i+1); scanf(&quot;%i&quot;,&amp;numbers[i]);
        }
    }</source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">if</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ if2 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/if2">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/ConditionalStructure"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/js_source_ontology/For2"/>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">if(numbers[i] &gt; max) {
    max = numbers[i];
}</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ max -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/max">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/Variable"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/js_source_ontology/else_if"/>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">var max = 0</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ numbers -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/numbers">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/Variable"/>
        <has_type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/array"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
    ↪ string">numbers</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">int numbers[3]</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ option -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/option">
        <rdf:type rdf:resource="https://semantic-web.netlify.
```

```
↪ com/js_source_ontology/Variable"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
↪ com/js_source_ontology/var"/>
        <source_code rdf:datatype="http://www.w3.org/2001/
↪ XMLSchema#string">var option = 0;</source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/
↪ XMLSchema#string">option</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ prompt -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
↪ netlify.com/js_source_ontology/prompt">
        <rdf:type rdf:resource="https://semantic-web.netlify.
↪ com/js_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
↪ netlify.com/js_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
↪ com/js_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
↪ string">scanf</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
↪ XMLSchema#string">option = prompt(&quot;Option&quot;)</
↪ source_code>
        <rdfs:label rdf:datatype="http://www.w3.org/2001/
↪ XMLSchema#string">prompt</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ var -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
↪ netlify.com/js_source_ontology/var">
        <rdf:type rdf:resource="https://semantic-web.netlify.
↪ com/js_source_ontology/DataType"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
↪ string">var</name>
        <rdfs:comment xml:lang="en">Javascript dynamic datatype
↪ </rdfs:comment>
        <rdfs:label xml:lang="en">var</rdfs:label>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
↪ write -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
↪ netlify.com/js_source_ontology/write">
```

```
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/js_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
    ↪ string">document.write</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">document.write(&quot;Please choose an
    ↪ option and press enter:\n&quot;);</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ write2 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/write2">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/js_source_ontology/While"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
    ↪ string">document.write</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">document.write(&quot;1. Read 3 numbers\
    ↪ n 2. Print the max\n 3.Exit\n&quot;);</source_code>
    </owl:NamedIndividual>



    <!-- https://semantic-web.netlify.com/js_source_ontology/
    ↪ write3 -->

    <owl:NamedIndividual rdf:about="https://semantic-web.
    ↪ netlify.com/js_source_ontology/write3">
        <rdf:type rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/SystemFunction"/>
        <is_included_in rdf:resource="https://semantic-web.
    ↪ netlify.com/js_source_ontology/else_if"/>
        <is_type_of rdf:resource="https://semantic-web.netlify.
    ↪ com/js_source_ontology/Void"/>
        <name rdf:datatype="http://www.w3.org/2001/XMLSchema#
    ↪ string">document.write</name>
        <source_code rdf:datatype="http://www.w3.org/2001/
    ↪ XMLSchema#string">document.write(&quot;\nMax= &quot; +
    ↪ max + &quot;\n&quot;);</source_code>
    </owl:NamedIndividual>
</rdf:RDF>

```

```


<!-- Generated by the OWL API (version 4.5.9.2019-02-01
    ↪ T07:24:44Z) https://github.com/owlcs/owlapi -->
```
Listing 4.2: Owl source code for JS source code defined in 2.2