B2- Synthesis Pool

B-ADM-144

SBML file parser

Systems Biology Markup Language and parsing

v1.0

EPITECH.
L'ECOLE DE L'INNOVATION ET DE L'EXPERTISE INFORMATIQUE

Requirement

my_strtowordtab_synthesis

Write a C file named *requirement.c* containing a function that will split a string into words. Separators are all **non-alphanumeric** characters.
The function will return an array, in which each cell will contain a string's (representing a word) address.
The last element of the array will be 0, which marks the end of the array.
The transmitted string is not to be modified in your function.

It should be prototyped the following way:

```
char **my_strtowordtab_synthesis(char const *str)
```

Only `malloc` and `free` are allowed from libC.
The rest of the project will not be corrected unless this requirement is fully functional (and rewritten).

The file must be placed at the root of your git repository.
It will be compiled with our main function, and our Makefile (the `-I` flag being empty).

SBML file parser

Systems Biology Markup Language and parsing

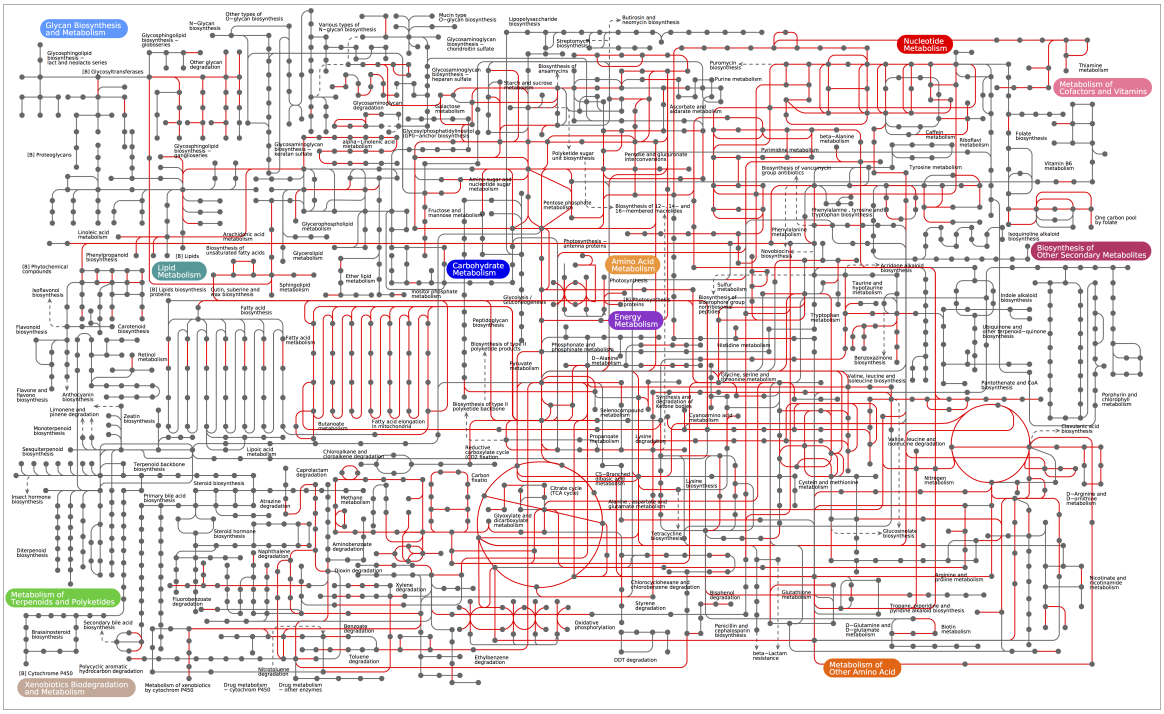
binary name:	SBMLparser
repository name:	ADM_SBMLparser_$YEAR
repository rights:	ramassage-tek
language:	C
group size:	1
compilation:	via Makefile, including re, clean and fclean rules

- Your repository must contain the totality of your source files, but no useless files (binary, temp files, obj files,...).
- All the bonus files (including a potential specific Makefile) should be in a directory named *bonus*.
- Error messages have to be written on the error output, and the program should then exit with the 84 error code (0 if there is no error).

Authorized functions: every functions from the libC

Many chemical reactions are responsible for living beings'' ability to develop, move, heal, communicate; digestion, for instance, is a chain of chemical reactions. 7439 chemical reactions are present in human beings.
A chemical reaction consists in transforming one or several *reactant(s)* into one or several *product(s)*. Molecules often have crude names, and are either represented by their chemical formulas or by acronyms; and their reactions are represented by equations such as:

$$C_6H_{12}O_6 + 6O_2 \rightarrow 6CO_2 + 6H_2O + ATP$$
$$ATP + GLCD \rightarrow ADP + G6P$$



Thus, one can demonstrate how a human being functions through the linking of chemical reactions that happen in their cells. This network is called the *metabolic network*, and is so complex (as you can see above) that it is a challenge for modern biology.

In order to organize this huge amount of information, a XML-based format, which uses specialized tags for metabolic networks, was created: the **SBML** format.
The main tags are:

1. *‹compartment›* to define the compartment of the reaction,
2. *‹species›* to define a chemical product,
3. *‹reaction›* to define a chemical reaction,
4. *‹speciesReference›* to refer to a previously defined chemical reaction,
5. *‹listOfCompartments›*, *‹listOfSpecies›*, *‹listOfReactions›*, *‹listOfReactants›*, *‹listOfProducts›*, which names are self-exlanatory.

If ever you are interested, you can read the full specifications here.

As an example, here is the SBML file that describes the second aforementioned reaction named: `example.sbml`:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<sbml xmlns="http://www.sbml.org/sbml/level3/version1/core" level="3" version="1" >
  <model name="Homo sapiens Glycolysis" id="Pathway146" >
    <listOfCompartments>
      <compartment name="Homo sapiens, Cell, Cytosol" id="Cytosol"/>
    </listOfCompartments>
    <listOfSpecies>
      <species compartment="Cytosol" name="Adenosine diphosphate" id="Compound1034"/>
      <species compartment="Cytosol" name="Glucose 6-phosphate" id="Compound1083"/>
      <species compartment="Cytosol" name="Alpha-D-Glucose" id="Compound1851"/>
      <species compartment="Cytosol" name="Adenosine triphosphate" id="Compound414"/>
    </listOfSpecies>
    <listOfReactions>
      <reaction reversible="false" name="Hexokinase-3" id="Reaction1232" >
        <listOfReactants>
          <speciesReference stoichiometry="1" species="Compound1851"/>
          <speciesReference stoichiometry="1" species="Compound414"/>
        </listOfReactants>
        <listOfProducts>
          <speciesReference stoichiometry="1" species="Compound1083"/>
          <speciesReference stoichiometry="1" species="Compound1034"/>
        </listOfProducts>
      </reaction>
    </listOfReactions>
  </model>
</sbml>
```

1- Simple parser

Write a program that will print, in alphabetical order, the list of tags and attributes found in the SBML file given as argument.
Each tag and each attribute will be unique.

Don't bother with error management concerning the tags in the SBML file. You'll get only well formatted files.

2- Information extraction

Add the *-i ID* option to your program.
If ID is the id of a `<compartment>` tag, the program will print an alphabetical list of id-associated chemical products on the standard output.
If ID is the id of a `<species>` tag, the program will print a quantified and alphabetical list of the chemical reactions that consume a chemical product on the standard output.
If ID is the id of a `<reaction>` tag, the program will print an alphabetical list of the reactants, and products of the reaction, on the standard output.
Otherwise, the program will print the list of all chemical products.

The *stoichoimetry* tag gives the quantity of consumed chemical products.

Add the *-e* option to print the equation of the reaction, if ID is a reaction id.

If the reaction is reversible, the arrow is "<->".

3- SBML to JSON

Add the *-json* option to convert the SBML file into JSON.
Tags and attibutes taken into account will be the one in the SBML file given in the previous example.

If the *-i* option is filled in, and:

1. if the id is a compartment, only this compartment, and the chemical products and reactions referring to it, will be displayed,
2. if the id is a chemical product, only this product, and the compartments and reactions referring to it, will be displayed,
3. if the id is a reaction, only this reaction, and the chemical products and compartments referring to it, will be displayed,
4. otherwise, the option is ignored.

Terminal

```
~/B-ADM-144> ./SBMLparser -h
USAGE
        ./SBMLparser SBMLfile [-i ID [-e]] [-json]

DESCRIPTION
        SBMLfileSBML file
        -i ID   id of the compartment, reaction or product to be extracted
                (ignored if uncorrect)
        -e      print the equation if a reaction ID is given as argument
                (ignored otherwise)
        -json   transform the file into a JSON file
```

Examples

Your program output has to be strictly identical to the one below.

Terminal

```
~/B-ADM-144> ./SBMLparser example.sbml
compartment
----->id
----->name
model
----->id
----->name
reaction
----->id
----->name
----->reversible
sbml
----->level
----->version
----->xmlns
species
----->compartment
----->id
----->name
speciesReference
----->species
----->stoichiometry
```

Terminal

```
~/B-ADM-144> ./SBMLparser example.sbml -i Cytosol
List of species in compartment Cytosol
----->Adenosine diphosphate
----->Adenosine triphosphate
----->Alpha-D-Glucose
----->Glucose 6-phosphate
```

Terminal

```
~/B-ADM-144> ./SBMLparser example.sbml -i ChuckNorris
List of species
----->Adenosine diphosphate
----->Adenosine triphosphate
----->Alpha-D-Glucose
----->Glucose 6-phosphate
```

Terminal

```
~/B-ADM-144> ./SBMLparser example.sbml -i Compound414
List of reactions consuming species Compound414 (quantities)
----->Reaction1232 (1)
```

Terminal

```
~/B-ADM-144> ./SBMLparser example.sbml -i Reaction1232
List of reactants of reaction Reaction1232
----->Compound1851
----->Compound414
List of products of reaction Reaction1232
----->Compound1034
----->Compound1083
```

Terminal

```
~/B-ADM-144> ./SBMLparser example.smbl -i Reaction1232 -e
1 Compound1851 + 1 Compound414 -> 1 Compound1034 + 1 Compound1083
```

Terminal

```
~/B-ADM-144> ./SBMLparser example.sbml -i Compound414 -json
{
     "listOfCompartments":  [
         {
            "id":  "Cytosol",
            "name":  "Homo sapiens, Cell, Cytosol"
         }
     ],
     "listOfSpecies":  [
         {
            "compartment":  "Cytosol",
            "id":  "Compound414",
            "name":  "Adenosine triphosphate"
         }
     ],
     "listOfReactions":  [
         {
            "id":  "Reaction1232",
            "name":  "Hexokinase-3",
            "reversible":  "false"
         }
     ]
}
```

Terminal

```
~/B-ADM-144> ./SBMLparser example.sbml -i Reaction1232 -json
{
      "listOfReactants":  [
          {
             "species":  "Compound1851",
             "stoichiometry":  "1"
          },
          {
             "species":  "Compound414",
             "stoichiometry":  "1"
          }
      ],
      "listOfProducts":  [
          {
             "species":  "Compound1083",
             "stoichiometry":  "1"
          },
          {
             "species":  "Compound1034",
             "stoichiometry":  "1"
          }
       ]
}
```