

Compiler Construction (CS 402)

Spring 2021 – Assignment 1

Due: March 07, 2021

Given below a JSON object, generate a tail end recursive program in C, C++, C# or Java only. JSON object is provided in a file in current folder called Formula.JSON. Output code is generated in a file in current folder with the name Code.C or Code.CPP or Code.CS or Code.JAVA

```
{
  "function": <<function name>>,
  "return-type": <<return type of the function>>
  "parameters": [
    {"parameter": <<parameter name>>, "type": <<parameter type>>},
    {"parameter": <<parameter name>>, "type": <<parameter type>>},
    {"parameter": <<parameter name>>, "type": <<parameter type>>},
    ... ..
    ... ..
  ],
  "conditions": [
    {"condition": <<condition expression>>, "action": <<action expression>>},
    {"condition": <<condition expression>>, "action": <<action expression>>},
    {"condition": <<condition expression>>, "action": <<action expression>>},
    ... ..
    ... ..
    {"condition": "", "action": <<action expression>>} // tail-end condition is blank
  ]
}
```

As an example, a factorial JSON object could be something like:

```
{
  "function": "factorial",
  "return-type": "int",
  "parameters": [
    {"parameter": "n", "type": "int"}
  ],
  "conditions": [
    {"condition": "n <= 0", "action": "1"},
    {"condition": "", "action": "n * factorial(n - 1)"},
  ]
}
```

This will result in

```
int factorial(n) {
    int temp;

    if(n <= 0) temp = 1
    else temp = n * factorial (n - 1);

    return temp;
}
```

Few more examples:

$$fib(n) = \begin{cases} 0 & , n = 0 \\ 1 & , n = 1 \\ fib(n-2) + fib(n-1) & , n \geq 2 \end{cases} \quad search(n, arr, v) = \begin{cases} false & , n < 0 \\ true & , arr[n] = v \wedge n \geq 0 \\ search(n-1, arr, v) & , arr[n] \neq v \wedge n \geq 0 \end{cases}$$

$$palindrome(s) = \begin{cases} true & , len = 1 \\ s[0] = s[1] & , len = 2 \\ s[0] = s[len-1] \wedge palindrome(substr(s, 2, len-2)) & , len > 2 \end{cases}$$

substr is defined as substr(string, position, characters), where position starts from 1