FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA

Institute for Informatics
System Software Group
Prof. Dr. Clemens Grelck
Advanced Functional Programming
November 2, 2023

Assignment 1

Elementary Programming in CiviC

This assignment series is supposed to familiarise yourself with the CiviC model language and to provide you with a test suite of example programs for your own CiviC compiler.

Assignment 1.1: CiviC Core Programming

Implement a CiviC module `core.cvc` that exports the following functions:

- `gcd(`*a*`,`*b*`)`
  returns the greatest common denominator of *a* and *b*;
- `fac(`*n*`)`
  returns the factorial of *n*;
- `fib(`*n*`)`
  returns the *n*-th Fibonacci number;
- `isprime(`*n*`)`
  returns `true` if *n* is a prime number and `false` otherwise.

Assignment 1.2: CiviC Nested Functions and I/O

Implement a CiviC module `coreio.cvc` that exports the following functions:

- `fibs(`*n*`)`
  print first *n* Fibonacci numbers;
- `primes(`*n*`)`
  print first *n* prime numbers.

The function `fibs` must make use of the function `fib` from the `core.cvc` module. In contrast, the function `primes` shall have a clone of function `isprime` as a nested local function definition.

Assignment 1.3: CiviC Arrays

Implement a CiviC module `array.cvc` that exports the following functions:

- `printIntVec( int[`*n*`] `*vec*`)`
  `printFloatVec( float[`*n*`] `*vec*`)`
  print *vec* to stdout;
- `printIntMat( int[`*m*`,`*n*`] `*mat*`)`
  `printFloatMat( float[`*m*`,`*n*`] `*mat*`)`
  print *mat* to stdout;
- `scanIntVec( int[`*n*`] `*vec*`)`
  `scanFloatVec( float[`*n*`] `*vec*`)`
  scan *vec* from stdin;
- `scanIntMat( int[`*m*`,`*n*`] `*mat*`)`
  `scanFloatMat( float[`*m*`,`*n*`] `*mat*`)`
  scan *mat* from stdin;
- `matMul( float[`*m*`,`*n*`] `*a*`, float[`*o*`,`*p*`] `*b*`, float[`*q*`,`*l*`] `*c*`)`
  multiply two floating point matrices *a* and *b* and store result in *c*;
- `queens( bool[`*m*`,`*n*`] `*a*`)`
  solve the well known 8-Queens problem (bonus challenge).

Note:

In the absence of characters and character strings in CiviC, your formatting options are very limited. Make the best out of it.

Note:

All above CiviC modules **must not** export a `main` function. For testing purposes write separate modules that do contain `main` functions with suitable but minimal test code and submit them alongside.

Note:

Assignment Series 1 must be submitted invidiually.

Assignment due date: Monday, November 13, 2023