Introduction to Languages for Scientific Computing

Prof. **Paolo Bientinesi**

pauldj@aices.rwth-aachen.de

Sequence

Input: n: integer
Output: *True* or *False*

1. Compute f, the prime factorization of n.
   Ex.: $n = 48234496 \rightarrow f := 2^{21} \times 23; \quad n = 1425 \rightarrow f := 3 \times 5^2 \times 19$
   Note: make sure the primes are sorted from smallest to largest.

2. "Flatten" f in g: convert the exponent into a factor.
   Ex.: $f = 2^{21} \times 23 \rightarrow g := 2 \times 21 \times 23; \quad f = 3 \times 5^2 \times 19 \rightarrow g := 3 \times 5 \times 2 \times 19$

3. Compact g into n': replace the multiplication with concatenation.
   Ex.: $g = 2 \times 21 \times 23 \rightarrow n' := 22123; \quad g = 3 \times 5 \times 2 \times 19 \rightarrow n' := 35219$

4. If $n' = n$ then return $isPrime(n)$.
   If $n' \neq n$ then $n := n'$ and go to step 1, for a maximum of 15 iterations.
   After 15 iterations, return $isPrime(n')$.

Challenge #4

Define the functions
`seqFun[yourLastName][n_]` and `seqRule[yourLastName][n_]`
that implement the procedure “Sequence” described in Page 2.

- `seqFun` can use user defined functions, but no replacement rules.
- `seqRule` can use replacement rules, but no user defined functions.
- All Mathematica internal functions are allowed.
- **Objective**: correct, clean (as much as possible) code.
- **Rule**: In the case of a tie, the earlier submission wins.

Submission

- Individual assignment.
- Prepare a Mathematica notebook named <yourLastName.nb>, containing the definitions for `seqFun` and `seqRule`.
- Include your name at the top of the notebook.
- Submission by email to `pauldj@aices.rwth-aachen.de`
- Email's subject: ‘`LSC-17 Challenge4 <your last name>`’
- **Deadline: Monday, July 17, 23.59pm.**