

#### 01\_basic:

23 (big number of dims):  $\times$  COMPILER EXIT CODE expected=0, returned=-6

29 (eols everywhere :o):  $\times$  COMPILER TIMEOUT

#### 02\_functions:

13 (calling multiple functions):  $\forall \times$  INTERPRETER EXIT CODE expected=0, returned=53

25 (too many functions (test for symboltable)):  $\times$  COMPILER EXIT CODE expected=0, returned=1

#### 03\_statements:

#### 04\_expressions:

14 (test for implicit conversion):  $\forall \forall \times$  STDOUT

27 (divide priority (fail types)):  $\forall \times$  INTERPRETER EXIT CODE expected=0, returned=57

32 (random math expression test):  $\forall \times$  INTERPRETER EXIT CODE expected=0, returned=57

38 (variables, functions and too many integers and doubles and implicit conversion of return):  $\forall \times$  INTERPRETER EXIT CODE expected=0, returned=53

40 (double implicit conversion):  $\forall \forall \times$  STDOUT

42 (maximum for c integer):  $\forall \times$  INTERPRETER EXIT CODE expected=0, returned=51

#### 05\_conditions:

#### 06\_loops:

03 (nested while loop):  $\forall \times$  INTERPRETER TIMEOUT

18 (loop from one to 1250):  $\forall \times$  INTERPRETER TIMEOUT

19 (substr in loop):  $\forall \times$  INTERPRETER TIMEOUT

#### 07\_strings:

02 (string concat):  $\forall \forall \times$  STDOUT

#### 08\_booleans:

#### 09\_scopes:

#### 10\_unary:

#### 11\_base:

#### 12\_global:

#### 13\_cycles:

#### 42\_combination\_of\_extensions:

#### 66\_errors:

21 (colision of argument name and function name):  $\times$  COMPILER EXIT CODE expected=3, returned=0

253 (fail double): × COMPILER EXIT CODE expected=2, returned=1

305 (forum test 2): × COMPILER EXIT CODE expected=3, returned=1

96\_programs:

04 (add numbers from 1 to n): √ × INTERPRETER TIMEOUT

14 (reverse string function): √ × INTERPRETER EXIT CODE expected=0, returned=53

99\_benchmarks:

09 (multiple string concatenations): √√ × STDOUT

15 (torture test without extensions): × COMPILER EXIT CODE expected=0, returned=-11

RESULTS: 95.48% (444/465)