Question 3: Floats

Sean Marshallsay

The results of compiling the program with `gfortran-4.6.3` are shown in section 1 and the results of compiling the program with `ifort` are shown in section 2. When using 32-bit floating point representation there are 135 incorrect floating point calculations while there are only 82 incorrect calculations when using a 64-bit representation. This is because the mantissa only has a fixed number of bits so there are only a finite number of digits which can be stored, since real numbers can require an arbitrarily large number of digits to store accurately there will be numbers that can not be store. Increasing the number of bits available for floating point storage increases the size of the mantissa allowing more numbers to be accurately represented.

When the print statement on line 14 is present there is no change in results when the optimisation is changed, regardless of the number of storage bits or the compiler used. When the print statement is omitted then compiling the program with `-Ofast` causes it to find zero incorrect calculations (one would assume that the entire loop is optimised out), but no other optimisations affect the running of the program with either compiler.

1 gfortran

```
GNU Fortran (Ubuntu/Linaro 4.6.3-1ubuntu5) 4.6.3
Copyright (C) 2011 Free Software Foundation, Inc.

GNU Fortran comes with NO WARRANTY, to the extent permitted by law.
You may redistribute copies of GNU Fortran
under the terms of the GNU General Public License.
For more information about these matters, see the file named COPYING

PRECISION=KIND(1.0)
==================

-O0
---
 Found          135

-O1
```

```
---
 Found           135

-O2
---
 Found           135

-O3
---
 Found           135

-Ofast
------
 Found           135


PRECISION=KIND(1.d0)
====================

-O0
---
 Found            82

-O1
---
 Found            82

-O2
---
 Found            82

-O3
---
 Found            82

-Ofast
------
 Found            82
```

2 ifort

```
ifort (IFORT) 14.0.0 20130728
Copyright (C) 1985-2013 Intel Corporation.  All rights reserved.
```

```
PRECISION=KIND(1.0)
===================

-O0
---
 Found           135

-O1
---
 Found           135

-O2
---
 Found           135

-O3
---
 Found           135

-Ofast
------
 Found           135

PRECISION=KIND(1.d0)
====================

-O0
---
 Found            82

-O1
---
 Found            82

-O2
---
 Found            82

-O3
---
 Found            82

-Ofast
```

```
------
 Found              82
```