

Assignment 1: Legacy Fortran (25%)

FOREST FIRE WEATHER INDEX

This is an exercise in deciphering a legacy Fortran program and its associated documentation to derive a new program in a contemporary version of Fortran.

Consider the Fortran IV program described in the following technical report:

- Van Wagner, C.E., Pickett, T.L., “Equations and Fortran IV Program for the 1976 Metric Version of the Forest Fire Weather Index”, Petawawa Forest Experiment Station, (1975)
Available: <https://d1ied5g1xfp8x8.cloudfront.net/pdfs/23602.pdf>

The program is written in Fortran IV for a DEC PDP-8L. It generates a Forest Fire Weather Index based on the use of metric weather observations. The paper describes the following:

- All equations used to calculate the various values.
- A list of symbols used in the equations.
- A Fortran IV program to perform the calculations.
- A sample input and output.

The Fortran IV program, `ffwiFIV.for`, will be provided, in addition to the input file `sample.txt` (they can be found in the discussion group for A1). The Fortran file will compile and run, using redirected input, i.e.

```
> a.out < sample.txt
```

The sample output from this input file can be found at the end of this document.

TASK

Using the information found in the technical report and the program provided, design and implement a modern Fortran program (i.e. \geq F95) which is capable of taking the input as provided, and producing appropriate output. Perform the following tasks/requirements:

- Create a Fortran module named “**FFWIndices**” containing a series of subprograms to perform the calculation of the various values.
- Create a main “wrapper” program named “**ffwi**” which contains the I/O subprograms, and calls the various subprograms from the module above.
 - The I/O subprograms should prompt the user for the appropriate I/O filenames.
 - Provide readable, formatted output in the output file.
- Document the program as required, providing documentation relating the equations used in the paper to the program.

- Improve the usability of the program, i.e. filename prompts, user feedback and output formatting.

COMPILING

Please do not include a Makefile, and make sure your program compiles in the following manner:

```
> gfortran -Wall FFWIndices.f95 ffwf.f95
```

SAMPLE I/O

As outlined in the paper. The input file is provided, as is a sample of the output from the original program.

SKILLS

- Fortran programming, re-engineering by translation, program comprehension.

ASSIGNMENT INFORMATION

REFLECTION REPORT

Discuss your program in a **one (1) page (single-spaced)** reflection report, explaining the decisions you made in the process of reengineering your program. It should provide a synopsis of your experience with Fortran.

You should attempt to answer the following questions:

- What were the greatest problems faced while re-engineering the algorithm in Fortran?
- What particular features make Fortran a good language?
- Given your knowledge of programming, was Fortran easy to learn?

DELIVERABLES

The submission should consist of the following items:

- The reflection report (PDF).
- The code (well documented and styled appropriately of course):
FFWIndices.f95 ffw.f95
- Both the code and the reflection report can be submitted as a ZIP, TAR, or GZIP file.

STYLING & COMMENTS

Style consists of mnemonic variable names, indentation, and the use of whitespaces and paragraphing. The purpose of good style is to make the meaning of your program clear to someone who has never seen it before, cannot run it, and cannot talk with you. Documentation consists of in-code documentation. Examples of qualities to look for include:

- Are variable names well chosen?
- Are comments relevant rather than simple repetitions of the code?
- Do comments point out key sections of code, indicate special cases, or make assertions?
- Are the indents 3 or 4 spaces? Do not use tabs or 2 space indenting (please check “convert tabs to spaces” in your editor)
- Is whitespacing used to separate parts of the program to provide clarity?

SKILLS

- Fortran programming, re-engineering, program comprehension, ability to review specifications, ability to add on additional functionality, ability to write a Fortran program.

SAMPLE OUTPUT

DATE	TEMP	RH	WIND	RAIN	FFMC	DMC	DC	ISI	BUI	FWI
4 23	3.9	98	14	8.9	25	3	4	0	2	0
4 24	7.2	34	18	10.7	55	2	2	1	2	0
4 25	12.2	23	11	0.0	78	5	5	2	4	1
4 26	10.0	28	14	0.0	86	6	8	5	6	4
4 27	21.7	22	13	0.0	92	11	13	10	10	10
4 28	22.8	42	13	0.0	91	14	18	10	14	11
4 29	15.0	62	5	0.0	89	16	22	4	15	6
4 30	9.4	67	6	0.8	82	16	24	2	16	3

DATE	TEMP	RH	WIND	RAIN	FFMC	DMC	DC	ISI	BUI	FWI
5 1	3.3	42	31	8.9	58	9	15	1	9	1
5 2	10.6	23	16	0.0	80	11	19	3	11	3
5 3	6.1	96	16	0.3	76	11	23	2	11	2
5 4	4.4	34	27	2.0	70	11	26	2	11	2
5 5	7.2	40	8	0.3	79	12	29	2	12	1
5 6	3.3	91	18	14.0	30	5	11	0	5	0
5 7	4.4	83	6	22.6	24	2	3	0	2	0
5 8	7.2	64	8	0.8	49	3	7	0	3	0
5 9	6.7	93	6	16.8	20	1	4	0	1	0
5 10	6.7	71	14	7.9	39	1	4	0	1	0
5 11	10.0	53	18	0.0	66	2	8	1	2	0
5 12	11.7	98	3	14.2	17	0	5	0	1	0
5 13	8.3	79	18	11.2	36	1	4	0	1	0
5 14	10.6	81	19	4.6	42	1	4	0	1	0
5 15	14.4	94	5	15.5	20	0	5	0	0	0
5 16	13.9	40	14	0.0	64	3	10	1	3	0
5 17	19.4	41	16	0.3	82	6	16	4	6	3
5 18	12.2	40	18	0.0	86	8	20	6	8	6
5 19	10.0	41	16	0.0	87	10	25	6	10	6
5 20	16.1	32	8	0.0	89	13	30	6	13	7
5 21	18.9	36	10	0.0	90	16	36	7	16	9
5 22	28.9	46	10	0.0	90	20	43	7	20	10
5 23	23.3	54	13	0.0	89	23	50	7	23	12
5 24	18.3	44	14	1.0	85	26	56	4	26	8
5 25	11.1	60	14	0.3	85	27	60	4	27	8
5 26	10.0	64	8	1.3	74	28	64	1	28	2
5 27	15.6	50	6	3.0	65	24	67	1	25	1
5 28	18.3	35	6	0.0	81	28	72	2	28	4
5 29	12.2	88	6	5.1	41	18	70	0	22	0
5 30	16.1	46	8	0.0	70	20	75	1	24	1
5 31	20.6	59	23	0.0	81	23	81	5	27	8