# Default

Kieron O'Connor

| | | | |
|---|---|---|---|
| | **COLLABORATORS** | | |

| | *TITLE* : Default | | |
|---|---|---|---|
| *ACTION* | *NAME* | *DATE* | *SIGNATURE* |
| WRITTEN BY | Kieron O'Connor | November 2, 2022 | |

**REVISION HISTORY**

| NUMBER | DATE | DESCRIPTION | NAME |
|---|---|---|---|
| | | | |

# Contents

# Chapter 1

# Default

## 1.1 FileComp V2.1 Documentation

FileComp V2.1 Documentation

© Kieron O'Connor 1995

Disclaimer
 : Legal mumbo jumbo.

Overview
 : What this program is.

Requirements
 : What you need to run it.

Installation
 : How and where to install it.

Usage
 : How to run it.

DOpus Usage
 : How to run it from DOpus.

Hints & Tips
 : How to get the best out of it.

Stats
 : How fast is it.

Limitations
 : What it doesn't do.

Bugs
 : What problems are there.

Author
 : About me.

```
History
 : What the program has been through.

Thanks
 : Who helped.

Index
 : Quick access to each part of this file.
```

## 1.2  Disclaimer

This program is Freeware but the copyright for the code is still held
by myself. You allowed to copy, distribute, delete, archive, pack this
program if you so wish. I take no responsibility for the stability of
this program, you are using it at your own risk . If this program
crashes and you harddisk is trashed then you are on your own, I am not
responsible.

USE AT YOUR OWN RISK!!!

## 1.3  Overview

FileComp is a utility that will compare 2 or more files and report the
results. That is all this program basically does. It will let you choose
whether you want to show what locations are the same, whether you want the
output in decimal, hexadecimal or octal and whatever else I can add.

I wrote this so that my brain wouldn't turn to mush while I was on the
summer break. It also gave me something to do when I was bored.

## 1.4  Requirements

An Amiga. You should have one of these. If not, buy one now.

WB2+. If you don't have this, WHY????

Some files to compare. Why else would you need a compare utility.

Directory Opus (DOpus) would also be nice. A program that should be
included with every Amiga.

As much memory and as fast a processor as you can afford. You can never
have too much.

Also a fast storage device. Always a good idea.

## 1.5  Installation

This program is very easy to install. Ideally it should be  ↩
installed in
the C: directory. To do this, click here.

The reason for this is so that  you have access to the program no matter
where you are.

If you don't want to install it in C:, then feel free to copy the program
to where ever you like.

The program only accesses libraries that are contained in ROM so you
shouldn't have any problems. If you do, then please contact me at my

e-mail
 address.

## 1.6  Usage

The program is designed to be used from the Shell.

These are the supported command line options:

FileComp
FILE1/A
,
FILE2/A
,
FILE3
,
FILE4
,
FILE5
,
FILE6
,
FILE7
,
FILE8
,

D=DEC/S
,
O=OCT/S
,
SM=SAME/S
,
NZ=NOZERO/S
,
FI=FILEINFO/S
,

MED=MEDIUM/S

```
                        ,
                        LRG=LARGE/S
                        ,
                        Q=QUIET/S
                        ,
                        V=VERBOSE/S
                        :
```

The defaults for the program are as follows:

Comparison:  Differences. The differences between the file are shown.
Buffer size: Small. A small buffer is used. The size is 1024 bytes.
Output type: Hexadecimal. The ouput is printed in hexadecimal.

If you don't know why there is an = sign. This is for a shortcut
for that option. This means you can use D instead of DEC or O
instead of OCT.
The /A means that there has to be a value for that option to work.
The /S means that it is a switch. If you enter that option, the switch
is on otherwise it is off.

FILE1/A,FILE2/A,FILE3,FILE4,FILE5,FILE6,FILE7,FILE8

These are for the files that you want to compare. Upto 8 files are
supported. There has to be at least 2 files for the program to work.

D=DEC/S

This makes all output appear in decimal.

O=OCT/S

This makes all output appear in octal.

SM=SAME/S

This will tell the program to display the location and value where
all the entered files are equal.

NZ=NOZERO/S

When used in conjunction with SAME/S, this will not print out any
location where all the values are zero.

FI=FILEINFO/S

This will display a little information about each file. At present
it is only the filesize.

MED=MEDIUM/S

This will set the memory buffer for each file to MEDIUM or 2048 bytes.
This is a good comprimise of speed to memory.

LRG=LARGE/S

This will set the memory buffer for each file to LARGE or 8192 bytes.

This gives the a boost to performance. This is the one I personally suggest if you have the memory to spare.

```
Q=QUIET/S
```

This will only display the total number of occurrences found during the comparison. This is useful for a quick check of large files.

```
V=VERBOSE/S
```

This will print out extra information for each value found. This does make the program slower because of the extra information that has to be displayed.

## 1.7  DOpus Usage

It is useful to have a button set up in DOpus to compare files. The following settings are the ones I use.

Button Name FileComp (This can be anything you like.)

AmigaDOS   C:FileComp {O} FileInfo

Then set the flags to:

```
CD Source
Output Window
```

and the Close Delay to -1.

## 1.8  Hints & Tips

When dealing with large files, use the quiet switch first to check on how many occurrences there will be. This may save you a lot of time.

Use a program like KinCON to give you Shell a scroll bar so that you can see info that has just flown off the top of the screen.

OR

Use '>' to re-direct the output to a file. If you don't know how to do this then read on. It is quite easy. The format is:

```
command > filename
```

That is it. So to re-direct to a file in RAM: called FileCompList, you would enter the following:

```
FileComp AnyFile1 AnyFile2 > RAM:FileCompList
```

This will then create a file in RAM: called FileCompList. This file will contain all the output that would have gone to the screen.

If possible, have the Shell windows height as small as possible. This
will help with the speed. It also helps if you have KingCON installed.

## 1.9   FileComp Stats

This program has been tested only on my personal system. If it doesn't
work on yours then please get in touch and tell me what the problem
is. I will then attempt to rectify the problem.

Here are some stats on the program itself.

It was written in Modula 2 and C.
The Modula 2 source totals 14651 bytes including comments.
The C source totals 3096 bytes.

The compiled size with array, pointer, range, stack, div by 0 checking
is 12288 bytes. This can drop to 11452 bytes if I remove these.

The setup used to test the program is as follows:

Amiga 1200 with a Blizzard 1230/III 50 with 50 mhz FPU. This has 8 meg
of Fast RAM on board. There is also an internal 2½ inch 250 meg Seagate
HDD.

With this setup, I get about 50k per second out of the program. This will
vary depending on the amount of screen output there is. The more output,
the slower the program.

I can speed this up by using a larger buffer for each file. At present
this is set to 2048 bytes. This may change in the future.

## 1.10   Limitations

There are some limitations to the program. They are listed below.

Will only compare 8 files at a time.

Careful when using DosTrace V2.20. It doesn't agree with the program.

DosTrace will also cause problems with KingCon and some other programs.

## 1.11   Bugs

These are the bugs I have found and have hopefully fixed. If you find
any then please contact me and I will attempt to fix them.

Version 1.X

Problem with the screen output. It doesn't display enough spaces between

```
each value. 5/8/95.


Version 1.2

Missing out last item in file. 6/8/95.


Version 1.4

If you repeat same/diff, only the last one is relevant.
If the 1st and last filenames are the same, only these 2 files are
checked. 7/8/95.
If you enter one file, the whole file is displayed. 8/8/95


Version 1.6

I am writing blank spaces that are not needed. 8/8/95
If no file is entered but everything else is okay, then you get dumped
out of the program. 9/8/95.


Version 2.0

The comparefiles C module is destroying the array.
Have created a work around. 15/9/95.
C module won't print out Hex properly.
C module destroying array when using same switch. 16/9/95.
System crash if you select a dir instead of a file. 18/9/95.
Out of range on the WriteOccur module. 19/9/95.
```

## 1.12  History

```
Just a brief history of what I have done to the program.

Version 1.0

Started on 18/7. Only wrote a couple of lines of code.
Returned to it on the 5/8. Finished coding that day.


Version 1.1

Added option to display either the differences or the same values in
each file. 6/8/95.


Version 1.2

Fixed bug where I missed the last item in a file. 6/8/95.


Version 1.3
```

Wrote a proper argument parser. Now you have to use diff or same
instead of d or s. Some internal changes. 7/8/95.


Version 1.4

Rewrote large amount of code. Now handles more that 2 files at the
same time. 7/8/95.


Version 1.5

Fixed bug where if you repeat same/diff, only the last one is relevant.
Fixed bug where if the 1st and last filenames are the same, only these
2 files are checked.
Fixed bug where if you enter one file, the whole file is displayed. 8/8/95


Version 1.6

Added new command line option that allows user to choose whether the
output is dec, hex or oct.
Fixed bug that kept screwing around with the amount of spaces to be
inserted between each value.
Fixed bug where I am writing too many spaces. 8/8/95.


Version 1.7

Fixed bug where you got dumped out of the program if you didn't enter
a filename. 9/8/95.


Version 1.8

Complete rewrite of code. Commandline now accepts 4 files, has a set
of defaults built in so that if no options are entered, it will continue,
the last option of each group is the one that is used, different file
access routines, added option for byte,word,long to decide on the size of
each search. Still haven't gone to V2 because of bugs that will
appear. 12/8/95.


Version 1.9

Can't remember much about this version. I will have to check the source
code and see what I did.


Version 2.0

Removed Byte/Word/Long. It wasn't needed. FileIO speeded up.
Bug removed with compare procedure. (15/8/95)
Added Quiet and NoZero switches. (16/8/95)
Renamed program to FileComp. (17/8/95)
Added Verbose switch. (18/8/95)
Increased number files to 8.

Added in switches to compare each file with a different parameter.
Changed commandline, it was too long. (19/8/95)
Removed switches to compare each file. now just global switches.
Created GUI, will attempt to implement.
Rewrote file access. Program now smaller.
Will give standard DOS error numbers. (23/8/95)
Rewrote screen access. Now in C module.
Added error strings.
Program now smaller. (14/9/95)
Rewrote compare part in C. Program now smaller.
Added more error strings.
Fixed bug where array was destroyed. (15/9/95)
Fixed array bug again. It was because I used a = (for assign) instead
of == (is equal to).
Rewrote print procedure in Mod2. Fixed bug writing hex.
Changed arg.filename into array. Allows easier checking.
Lots of minor code modification to make it use less memory. (16/9/95)
Changed file access from FRead to Read.
Added own file buffer. Program now far quicker and faster than Compare.
Done some time tests, they are below.

FileSize 1806051 bytes, 9 differences, 1st 3, 3 in middle and the last
3 bytes.

```
Buffer              Time
1024                40.64 secs
2048                34.29 secs
4096                same
8192                29.99
```

About 50k per second on this setup with quiet switch on.
A bloody long time without quiet switch (Depends on how much is written
to the screen).
Fixed crash bug. It happened when you tried to open a device or
dir. (18/9/95)
Fixed range error bug.
Added FileInfo switch to show file sizes. (19/9/95)


Version 2.1

Changed memory access. Now allocates memory that is needed instead of
bulk wastage.
This change in memory allowed another change. You can now
set the file buffer. This is done at the command line.
Options are Small, Medium, Large.
Added some abbreviations to the command line to make it easier
entering options. (21/9/95)
Changed buffer sizes. They are now 1024, 2048 and 8192.
Changed number in C module. Now less prone to problems. (22/9/95)
Added more error strings. (25/9/95)
Reduced command line as some arguments were superfluous. (26/9/95)


## 1.13   Author

That is me, Kieron O'Connor. I am at present a student at Brunel University
studying  Computer Science. I can be reached at my e-mail address there
until 1997 when I will hopefully graduate. The e-mail address is below:

    cs93kjo@brunel.ac.uk

If you wish to write to me, I can be reached at:

    Kieron O'Connor
    56 Chandos Avenue
    Ealing
    London
    England
    W5 4ER

 All comments are welcome and will be replied to. You stand a better chance
 if you use e-mail though.

## 1.14  Thanks Go To

 These are the people that helped in the making of this program.

 Amritpal Mann for writing the Turbo Modula compiler that this program was
 compiled with.

 Matthew Dillon for his Dice C compiler. This was used to compile the C
 code and is also used by Turbo Modula.

 Michael Sinz for the Enforcer. This saved me many times.

 Dave Larsson for KingCON. This saved me much typing.

 Dietmar Eilert for GoldED. A very nice text editor.

 The original Amiga team for a rather good computer.

## 1.15  Index