C Programming

Divide numbers

Problem

- Write a program `div3` that reads in three floating point numbers, adds the first two and divides the total by the third number. Test with these data:

```
(1610 + 2004) / 2365
```

- Bonus program for extra credit points: change the program to `div4`. This program should read in four floating point numbers. Add the first two together, then add the second two together. Divide your first sum by your second sum. Test with these data:

```
(1610 + 2004) / (2005 + 360)
```

- Submit in Schoology as an Org file with your name and `#+HONOR: pledged` in the header. The code blocks should be run and the `#+Result:` should show the correct answer.
- To get full points, submit the Org file with some comments outside of the code blocks - explaining what you do, what the code is about, or anything else really that you want to share. Use NoppeLayoutModified.org as an inspiration.
- A comment could look this is: I absolutely, and unreservedly LOVE this class!
- This exercise originally came from Joyce (2018).

Solution

Meta data

The meta data can also be added as an Org-mode example block 1:

```
#+TITLE: Divide numbers
#+AUTHOR: Marcus Birkenkrahe
#+HONOR: Pledged
```

Add and divide 3 numbers

- We want to divide three floating point numbers x, y and z.
- The code in 1 reads test data in via the file `div3`
- The test data are formatted as `float` numbers

- The input file is created with a shell command in 1

```
echo "1610. 2004. 2365." > div3
```

```
float x,y,z, result;

scanf("%f %f %f", &x, &y, &z);

printf("(%.0f + %.0f)/%.0f = %f\n",
       x, y, z, result = (x+y)/z);
```

Add and divide 4 numbers

- In 1, we now read four numbers from the file `div4`
- The input file is created with a shell command in 1

```
echo "1610. 2004. 2005. 360." > div4
```

```
float x,y,z,u, result;

scanf("%f %f %f %f", &x, &y, &z, &u);

printf("(%.0f + %.0f)/(%.0f + %.0f) = %f\n",
       x, y, z, u, result = (x+y)/(z+u));
```

Truncation

- What happens if we accidentally declare the variables to be `int` rather than `float`?
- The code in 1 reads test data in via the file `divint3`
- The test data are formatted as `int` numbers
- The input file is created with a shell command in 1

```
echo "1610 2004 2365" > divint3
```

```
int i,j,k, result;

scanf("%d %d %d", &i, &j, &k);

printf("(%d + %d)/%d = %d\n",
       i, j, k, result = (i+j)/k);

printf("1610./4200000. = %f\n",1610./4200000.);
```

- We see that the result is truncated to 0, which is wrong.
- The last line gives the result that is computed, but it cannot be displayed as an `int`, only as a `float`.

References

- Joyce (2018). Numerical C. Springer Apress.
- Kernighan, Brian W.; Ritchie, Dennis M. (February 1978). The C Programming Language (1st ed.). Englewood Cliffs, NJ: Prentice Hall. ISBN 0-13-110163-3. URL: archive.org.
- Orgmode.org (n.d.). 16 Working with Source Code [website]. URL: orgmode.org

Author: Marcus Birkenkrahe

Created: 2022-03-17 Thu 15:46

Validate