

# Staffan's diary

## 2017-02-27

PDF-ify diaries (remove old text files), write shortcomings in documentation, organize final work with other team members. Slight restructure of documentation.

## 2017-02-24

5 not very productive hours

- Discuss ideas/implementation/program behavior with team members
- Add instructions for running the interpreter to documentation along with other smaller changes

## 2017-02-23

Approx. 2 hours

- Tweaking documentation
- Write Befunge programs and mess with interpreter (nothing committed)

## 2017-02-22

Approx. 3 hours (slow) work

- Fixed file extensions of Befunge programs (.bf -j .b93)
- Documentation: 2 more examples and various small changes

## 2017-02-20

- Separated initialization of memory from function main to simplify future testing.
- Continue work on documentation; description of the funge space and some examples.

## 2017-02-19

Added instruction table to documentation.

## 2017-02-18

Added better bounds-checking for BMemory.hs and began working on documentation.

## **2017-02-15**

Added basic structure to the Github repository (folders /befunge-programs, /diaries and /src) in an attempt to keep things organized. Comitted files (main.hs, BStack.hs, BMemory.hs, BArray.hs) that had been worked on by me (and partially Erik) before we set up the repository.

## **Patrik's diary**

### **2017-02-17**

Fixed some logical errors in the memory module, specifcly:

- Added constraint on height and width when importing from text file.
- Fixed constraint checking for getValue and putValue functions.

Also added and implemeted printAscii and duplicate as well as implemented setDirection

### **2017-02-19**

Located error in inputInt, will be fixed by Erik.

Added and implemented the functions modulo, logicalNot, greaterThan, ifHorizontal, ifVertical, swap and discard.

Also oranized functions so they apeare in the same order they do in the documentation for Befunge-93.

### **2017-02-24**

Added function specifications for some modules

### **2017-02-27**

Finalizing project with group. Added to documentation.

## **Erik's diary**

### **15/2 - 2017**

Today I did a couple of things to fix the overall structure to hopefully make it better. I noticed that we were declaring the same datatypes on multiple files. I tried to fix this by making another file called 'Types.hs' that would hold all datatypes in one place. Now of course, every file now needs to import this one if they need access some datatype. I also changed all imports in 'main.hs' to be qualified to reduce confusion, and name collisions.

## **16/2 - 2017**

Today I was asked by Staffan to merge his feature branch into master. It worked out smoothly, not too many merge conflicts or big changes. While i was at it i changed how the command '.' was programmed, moved the stepping och the programCounter to a better place to make some for now unimplemented commands alot easier to do and finally compressed how digits were read from the grid from 10 lines to just 1 by the help of the Data.Char module.

## **17/2 - 2017**

Today we did alot of things. All three of us had a long programming session together and we discussed alot of things about our program. For example how we should handle IO and whether the top function in BStack really is necessary or not. I also did a bunch of things on a new branch at home today. I added support for ?, , and &. I also made a bunch of small changes that in my opinion makes stuff better.

## **20/2 - 2017**

Today I added stringmode to ProgramCounter. It was easier than i first thought, i thought that i would need to change alot of pattern matches all over the place. But because we are smart with our program structure I only needed to change pattern matches in one file. I also ran hlint and ghc-mod on our files and got pages upon pages of suggestions and warnings. It took a while to fix all of them, but luckily most of the warnings were about unnecessary 'do's and '\$'s. I also fixed som small errors with the input handling. Now you can enter empty inputs without crashing and it should now work with all consoles. The error we had was that some consoles doesn't want to print its buffer until a newline character wanted to be printed. So I fixed that by calling a flush function after every 'putStrLn'.

## **21/2 - 2017**

Today I implemented the last two instruction-thingies, p and g. It took me about one hour. The essentials of the program are therefore done, so only function definitions and extra features are left. If we have time we consider adding flags so you can change the behaviour of the program without changing the source code.

## **22/2 - 2017**

Today I implemented flags so you can give the program options. It took me about 5 hours to do and i hope that everything i working as it should. I also took the time to write a help screen for the program. If you add the flag -help it will print out this screen. There it explains all program specific things that a user needs to know. Now there is only the boring things left, documentation and testing.

## **23/2 - 2017**

Today I did a couple of things. The first thing I did was to fix a bug Patrik discovered. The bug was that when the program was in stringMode and read a @ it exited instead of putting @ on the stack. The other thing I did today was to add colors to the output of the program. I also spent way too much time on trying to get the input prompt (*ii*) to disappear after an input was given. I got it to work mostly as I wanted it to. It splits the output into several lines and adds an extra newline if the cursor was on the beginning of a line, not perfect but good enough. I also tried to add parameters to change the width and the height of the array on the command line (so they are not just hardcoded). But I stopped halfway through because I noticed how annoying it would be to add that functionality. I would have to give MANY functions two additional arguments that they would rarely use. So I felt that it was too annoying and would make the code look too ugly, so I just gave up.

## **24/2 - 2017**

Today our group had very lengthy (1.5h-2h) discussion about how input should be handled. It was ultimately decided that it would be changed. So I spent maybe 30 min to change it. The color support I tried to add didn't work for some people, so we decided to remove it. To do that I did my very first git revert, it was pretty exciting :) Finally I added function definitions to Flags.hs

## **25/2 - 2017**

Added the last function definitions in Flags, Types, Main and BInstructions.

## **27/2 - 2017**

Today all three of us had a last minute working session on this project.