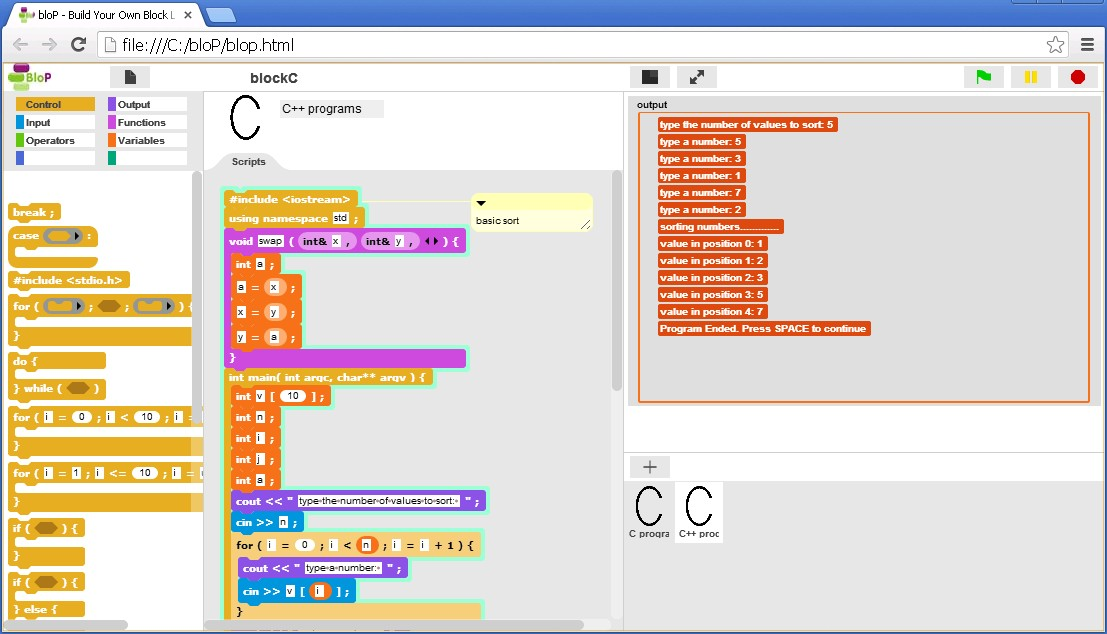
"bloP" is a development environment for block languages. The environment is a modification of Snap (<http://snap.brekeley.edu>). bloP stands for "BLOck Programming".

At the moment there are three languages available:

* blockC: minimal version of the C/C++ programming language
* blockLogo: minimal version of the Logo programming language
* blockASSL: animated language to describe search and sort algorithms

Every language is an XML file that contains both 1) the programming language and 2) the algorithms the user wrote by using that language.



Please, let me know of every improvement (both to the tool and the syntax of the programming language) you think I could make to the environment in order to make it really useful and easy-to-use for people who want to use block languages.

*WARNING:* bloP runs well either in Chrome or Opera but it doesn’t work in IE, Edge or Firefox.

**=== How to Startup a block language**

To startup the blockC language follow these steps:

1. open bloP.html either in Chrome or Opera

2. in File > Open select the blockC.xml file that you find in the “languages” folder (note: during loading your browser tells you that there is a script that is not responding just wait a little bit)

3. you can run the sample programs that you find in "C programs" or "C++ programs" or you can create your own blockC programs by adding a new program ("+" button, figure a) or duplicating a program (rightclick menu, figure b) or by using File > New (figure c)in order to create a completely new blockC program. Just snap together the C/C++ blocks you will find in the Input, Output, Control, Functions, Operators, and Variables categories.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| (a) | (b) | (c) |

4. save your programs by using File > Save As... The programs (and the language) will be saved in a new XML file that will be open in a new window. You can save the XML file by using the browser’s “save page” option.

**=== How to Modify or Create a block language**

To create or modify a new block language (for example to show/hide several blocks) just follow these steps (if you want to modify the blockC language load blockC in advance by following steps 1 and 2 above):

1. shiftclick the bloP logo and select "Unlock GUI" in the menu (it can take a while)

2. show the Snap primitives you need by rightclicking in the palette and selecting "show primitives" in the menu

3. make all icons visible in the corral by rightclicking the corral and selecting "add removed objects back to corral"

4. modify the environment as you like by:

- modifying custom blocks or adding new custom blocks

- renaming categories by righclicking them

- restoring default names and visibility for all categories by rightclicking the category frame

5. When you are done, go back to the bloP environment by shiftclicking the bloP logo and selecting "Lock GUI"

6. Save the language by using the File > Save As function and saving the XML file

NOTE: if you need a clean language file, that is without algorithms, just remove all algorithms from the bottom right area by rightclicking them and selecting “delete”.

**=== bloP Philosophy**

In order to make the bloP environment safe (that is, the user won't be allowed to modify the bloP environment when it is "locked": they won't be able to define new blocks, drag sprites, etc), so that when you write and run your programs you won't risk to impair the environment, Snap as been modified trying not to change the way Snap works (you should be able to load and run Snap XML files) but now you can do something in Snap (that is after unlocking BloP) that you can’t by using the standard version of Snap, that is:

* you can hide/rename categories (when you hide a category, all primitives of that category will be hidden) by rightclicking them
* you can hide the Stage and the sprites icons in the corral (but you need at least one visible sprite icon in the corral)
* you have 3 new Control blocks:

              - hat "when loading and never stop": the script under this hat is run when a program is loaded in Snap/bloP and it never stops running (even if the STOP Snap button at the top right corner is clicked)

              - hat "when running another script": the script under this hat is run before every bloP script is run (that is when you click on it or when you click the green flag)

              - block "run other scripts": this block, used at the end of a script that starts with the "when running another script" block, runs the script you intended to run when you clicked on it or clicked the green flag

* you can switch from Snap to bloP (and viceversa) by shiftclicking the logo and selecting "lock/unlock GUI".

When you lock the Snap GUI and switch to bloP you will have a different environment in which:

* Snap sprites/watchers on Stage are non draggable
* variable and list watchers cannot be modified
* by clicking the green flag button (at the top right corner) you don’t run all scripts (as it happens in Snap) but you only run the topmost script of the selected sprite
* you can't have programs (sprites) with the same name
* you don't see Snap sprite infos such as rotation style and draggability
* you can't hide/show “primitives” (that is Snap blocks)
* you can't hide/rename categories
* "sprites" are renamed "programs"
* you can't define or remove custom blocks
* you can't change settings
* you can't use the cloud storage
* costumes and sounds tabs are hidden
* the Snap button to create new sprites by drawing them is hidden

By shiftclicking the Snap logo you will be able to lock the GUI so that your language users won't be able to impair the environment by doing something wrong. But they will still be able to modify the environment, if they want, by shiftclicking the bloP logo and knowingly unlocking the environment.

**=== Create your own block language**

These features should allow you to develop your own programming language built by blocks.

By shiftclicking the bloP logo you will be able to unlock the GUI so that you will have the full power of Snap ([http://snap.berkely.edu](http://snap.berkely.edu/)) available to develop your own language blocks.

Use the "when loading and never stop" and "when running another script" blocks to define scripts that you need to run at setup time in order to prepare your environment ("when loading and never stop" hat) or to cleanup your environment each time a new program runs ("when running another script" hat). Remember: your users will be able to run a new program even before the running program has halted (the running program will be stopped).

bloP always runs in "locked" mode at startup so, when you are done, all you need to do is saving your language: there is no need to move back to bloP before saving. Furthermore, when your users will load your language in bloP they won't be able to impair the environment by doing something wrong.

**=== Comments are Welcome**

I'm very interested in hearing from you. The present design of bloP is mainly oriented to the implementation of text languages (such as C, Perl, Python, etc) but I'm continuously adding new features, so, if you have any suggestion about what you would need in order to implement your own block language (textual or visual), or you think that you had a good idea on how to improve bloP or you just found a bug, drop me a line at [sfederici@unica.it](mailto:sfederici@unica.it).

**=== Future Work**

I'm working on importing in bloP other block languages. If you want to collaborate to create an extensive library of block languages, please, do not hesitate to contact me at [sfederici@unica.it](mailto:sfederici@unica.it)

**=== Known Problems:**

- you can't use single quotes in block names (see "A" block in Operators category. It should be 'A' instead...)

- the comma after the last arguments of a function shouldn't be there

- grey rings (a feature of the Snap language) can be confusing to people that don't know what they are (this can happen to users of your own language that don't know Snap)

- switching between Snap and bloP is a bit slow (the whole program is saved and then reloaded).

- watchers on the Stage don't grow or shrink in “small Stage” or in “presentation mode”

- lists do not scroll to show the last added element

FINAL NOTE: bloP is based on Snap version 24/7/2013.