CONSOLE INTERFACE IMPROVEMENTS

DEEPAK CHETHAN (DODOCOCO)

ABOUT ME

- Name: Deepak Chethan (dodococo on telegram)
- Final year computer science undergrad from India
- Have been an Radare2 Contributor since Dec-2018
- Got selected to work on ”Console Interface Improvements”

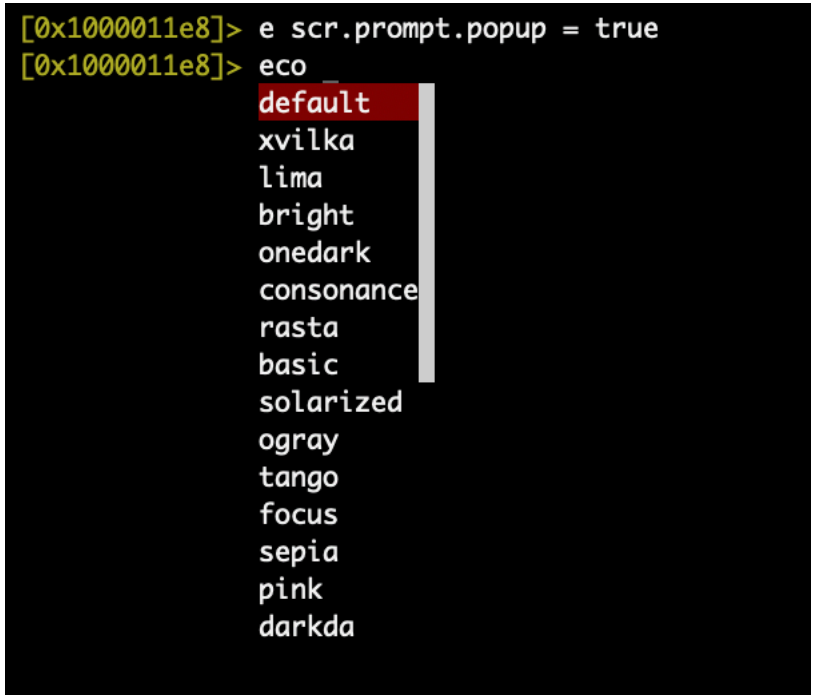
MAIN GOALS

- Generalize the code of popup widget
- Improving autocompletion and dietline modes (VI/Emacs-like hotkeys)
- Support colour scheme in radiff2 graphs
- Further improvements for UTF-8, RTL and Bi-Directional text
- Various bugfixes and improvements for visual, visual panels and graph modes
- Implement table commands and API

TASK 1: GENERALIZE THE CODE OF POPUP WIDGET

- Popup widget was already implemented in Visual Offset mode
- Refactored the code and integrated it into RCons dietline
- Can be enabled by setting the config variable scr.prompt.popup=true

HOW DOES IT LOOK?



TASK 2: AUTOCOMPLETION IMPROVEMENTS

- Mount shell (ms) was using its own version of autocompletion which is not complete
- Integrated the RCore autocompletion into mount shell
- Also added autocompletion for mount based commands like (mg, mc, mp, mo etc.,)

```
~ > r2 /bin/ls
 -- Don't look at the code. Don't look.
[0x1000011e8]> m posix /
Mounted posix on / at 0x0
[0x1000011e8]> ms
[/]>
ls       cd       cat      get      mount    ?        !        help     q        exit
[/]> cat
/./                                    /../                                   /home/                                 /Quarantine/
/usr/                                  /.Spotlight-V100/                      /net/                                  /.DS_Store
/.PKInstallSandboxManager-SystemSoftware/  /SAPDevelop/                       /bin/                                  /installer.failurerequests
/Network/                              /sbin/                                 /.file                                 /etc/
/var/                                  /Library/                              /System/                               /.fseventsd/
/private/                              /.DocumentRevisions-V100/              /.vol/                                 /Users/
/Applications/                         /opt/                                  /dev/                                  /Volumes/
/data/                                 /tmp/                                  /cores/                                /
[/]> cat
```

TASK 2: AUTOCOMPLETION IMPROVEMENTS

- Added SDB autocompletion (command k)

```
[0x1000011e8]> k
k         k?        ko        kd        ks        kj
[0x1000011e8]> k anal/
anal/fcns/      anal/meta/      anal/hints/     anal/types/     anal/spec/      anal/cc/        anal/zigns/     anal/classes/
[0x1000011e8]> k anal/spec/
anal/spec/gcc           anal/spec/spec.gcc.c    anal/spec/spec.gcc.d    anal/spec/spec.gcc.f    anal/spec/spec.gcc.g    anal/spec/spec.gcc.ld
anal/spec/spec.gcc.li   anal/spec/spec.gcc.lld  anal/spec/spec.gcc.lli  anal/spec/spec.gcc.llu  anal/spec/spec.gcc.lu   anal/spec/spec.gcc.p
anal/spec/spec.gcc.u
[0x1000011e8]> k anal/spec/
```

LIVE DEMO TIME

```
[0x1000011e8]> e scr.prompt.popup = true
[0x1000011e8]> eco default
[0x1000011e8]> ms
[/]> q
[0x1000011e8]> m posix /
Mounted posix on / at 0x0
[0x1000011e8]> ms /
[/]> cat /._
     /./
     /../
     /.Spotlight-V100/
     /.DS_Store
     /.PKInstallSandboxManager-Syst
     /.file
     /.fseventsd/
     /.DocumentRevisions-V100/
     /.vol/
```

```
[/]> q
[0x1000011e8]> k anal/types/_
               anal/types/*aligned_alloc
               anal/types/_Exit
               anal/types/__assert_fail
               anal/types/__assert_rtn
               anal/types/__bzero
               anal/types/__error
               anal/types/__libc_init
               anal/types/__libc_init_array
               anal/types/__libc_start_main
               anal/types/__maskrune
               anal/types/__stack_chk_fail
               anal/types/__tolower
               anal/types/__toupper
               anal/types/__uClibc_main
               anal/types/_exit
```

TASK 3: IMPROVING DIETLINE MODES

- Added support for unix commands like:
  - Uniq
  - Sort
  - Join
  - Head
  - tail

TASK 3: IMPROVING DIETLINE MODES

- Radare2 now support vi key bindings !!
- Can be enabled by setting scr.vi = true
- The current implementation supports:
  Insertion: *i, a, I, A, r and p*
  Movement: *h, l, 0, ^, $, w, b, e, W, B, E*
  Deletion: *d, dw, db, dB, dW, dh, dl, d$, d^, d0, D*
  c, cw, cb, cB, cW, ch, cl, c$, c^, c0 and x
  ciw/diw, cib/dib

TASK 3: IMPROVING DIETLINE MODES

- Integrated dietline into the HUD mode
- Now Hud supports all emacs/vi key bindings!
- Bonus: Added mouse support in HUD as well

TASK 4: RADIFF2 GRAPH IMPROVEMENTS

- Diff graphs in radare2 were super limited
- The only output it supported was xdot

TASK 4: RADIFF2 GRAPH IMPROVEMENTS

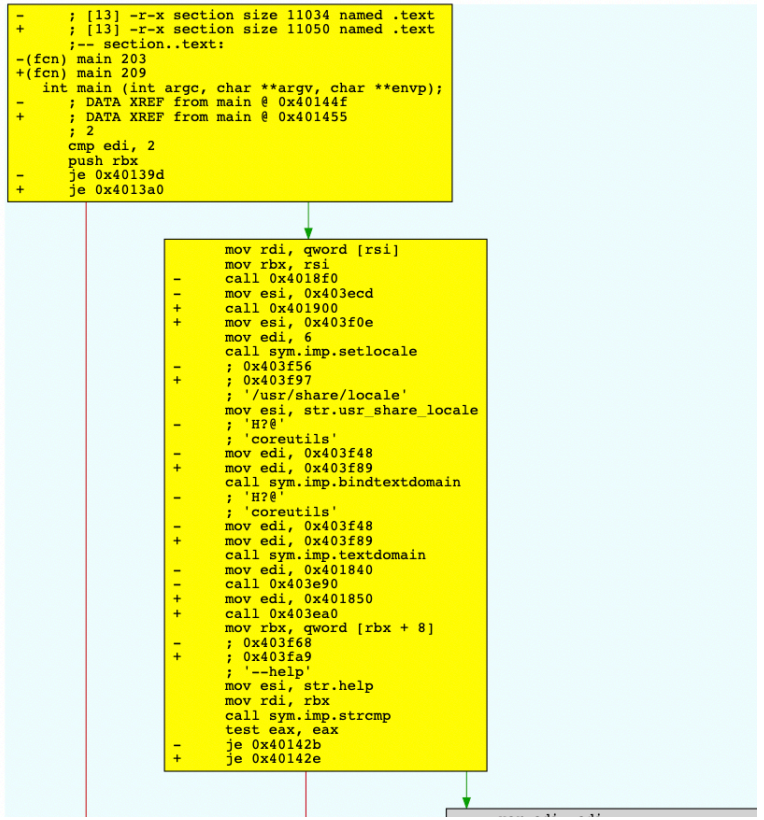
- So implemented more *agd* commands to construct diff graphs:
- Now it supports the output in:
  - r2 commands (*agd**)
  - Graphviz dot (*agdd*)
  - Graph Modelling Language (*agdg*)
  - Json (agdj)
  - SDB key-value (agdk)
  - Tiny ascii art (agdt)
  - Interactive ascii art (agdv)

TASK 4: RADIFF2 GRAPH IMPROVEMENTS

- Even radiff2 graphs supported only xdot output
- Now it even has interactive graphs (mini visual visual mode)
- Overall it supports all these commands

```
Graph Output formats: (-m [mode])
  <blank/a>  Ascii art
  s          r2 commands
  d          Graphviz dot
  g          Graph Modelling Language (gml)
  j          json
  J          json with disarm
  k          SDB key-value
  t          Tiny ascii art
  i          Interactive ascii art
```

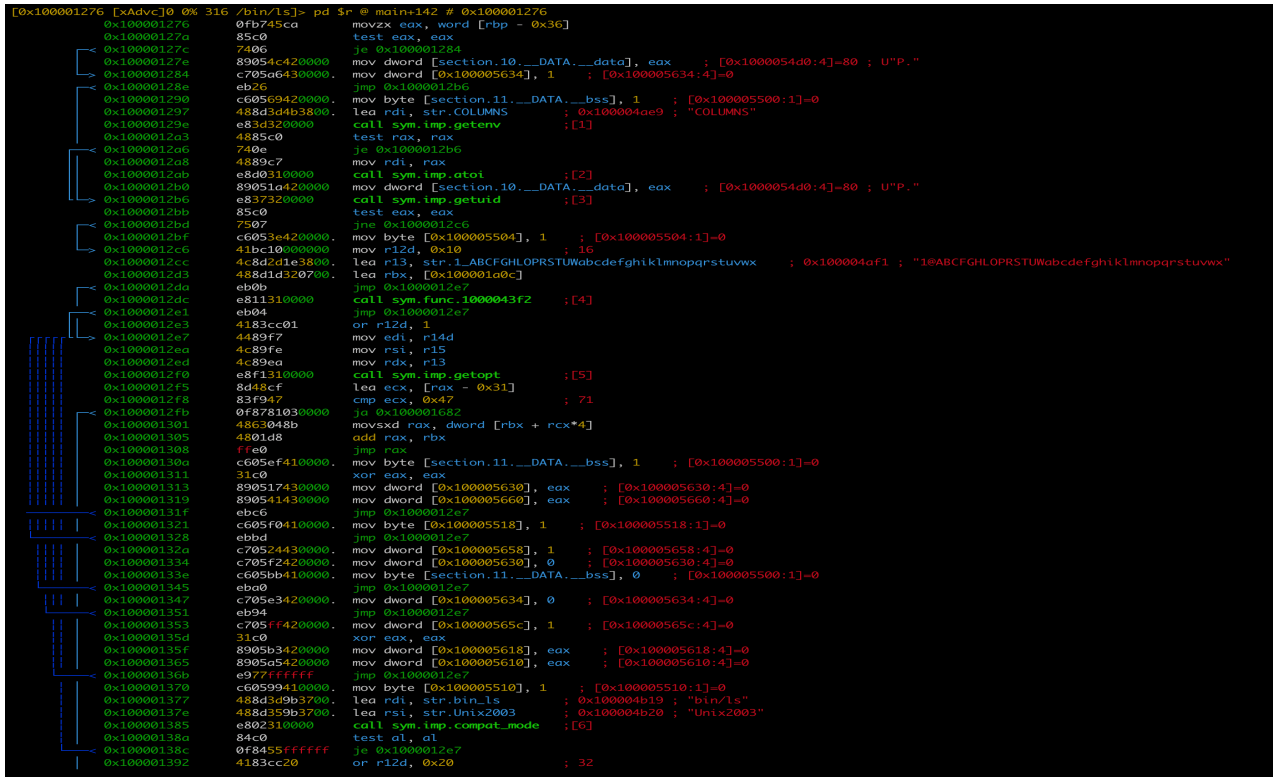
NEW ASCII ART GRAPHS VS OLD XDOT OUTPUT



TASK 5: UTF-8 BASED IMPROVEMENTS:

- All the reflines in disarm were of same color :|
- Enhanced it to have different colors for up and down arrows

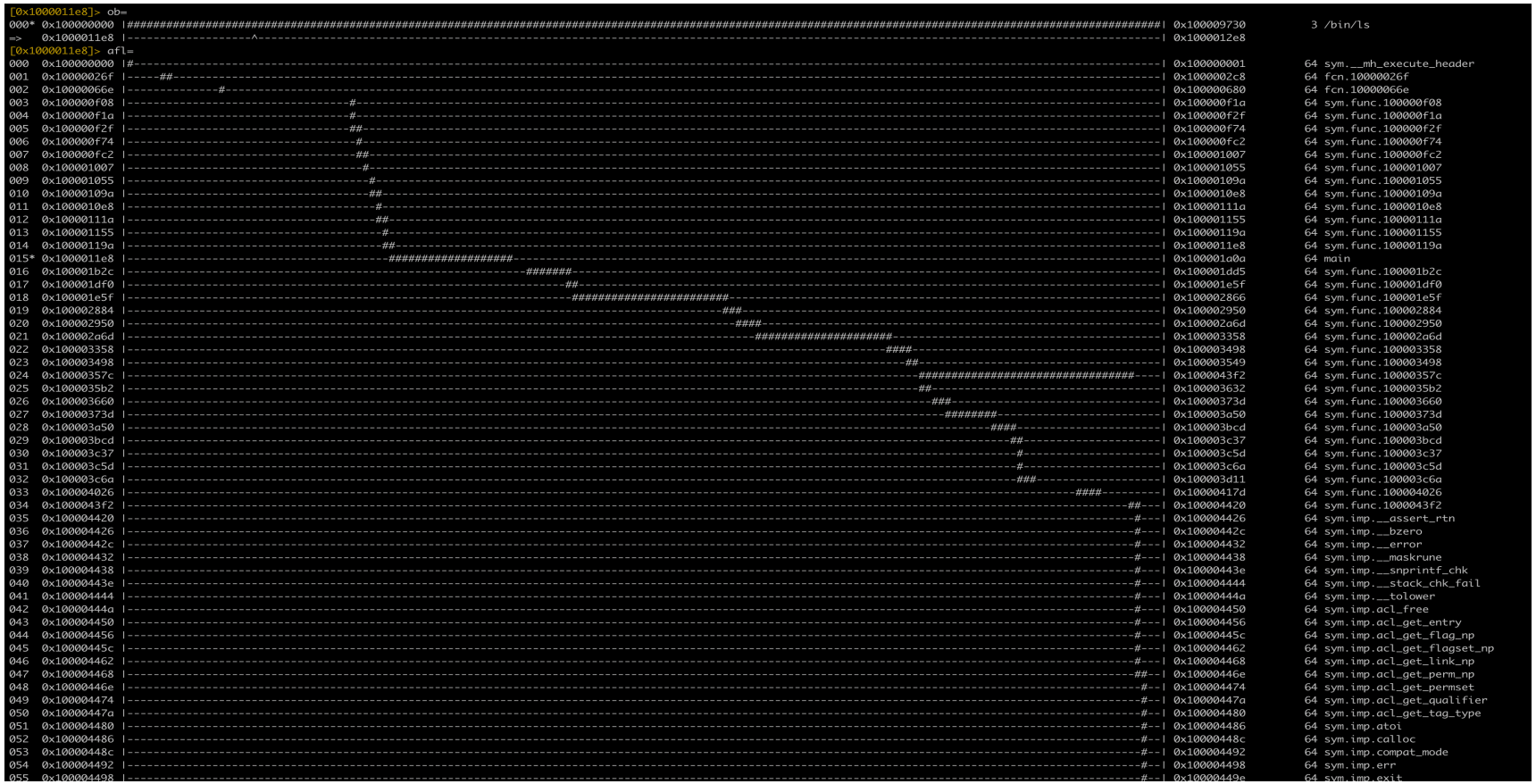
THE NEW DISARM



TASK 5: UTF-8 BASED IMPROVEMENTS:

- Implemented commands to show ascii-art colour bars with the section ranges
  - *ob=* (for open files)
  - *dt=* (Debug traces)
  - afl= (function lists)
  - *afb=* (basic blocks)

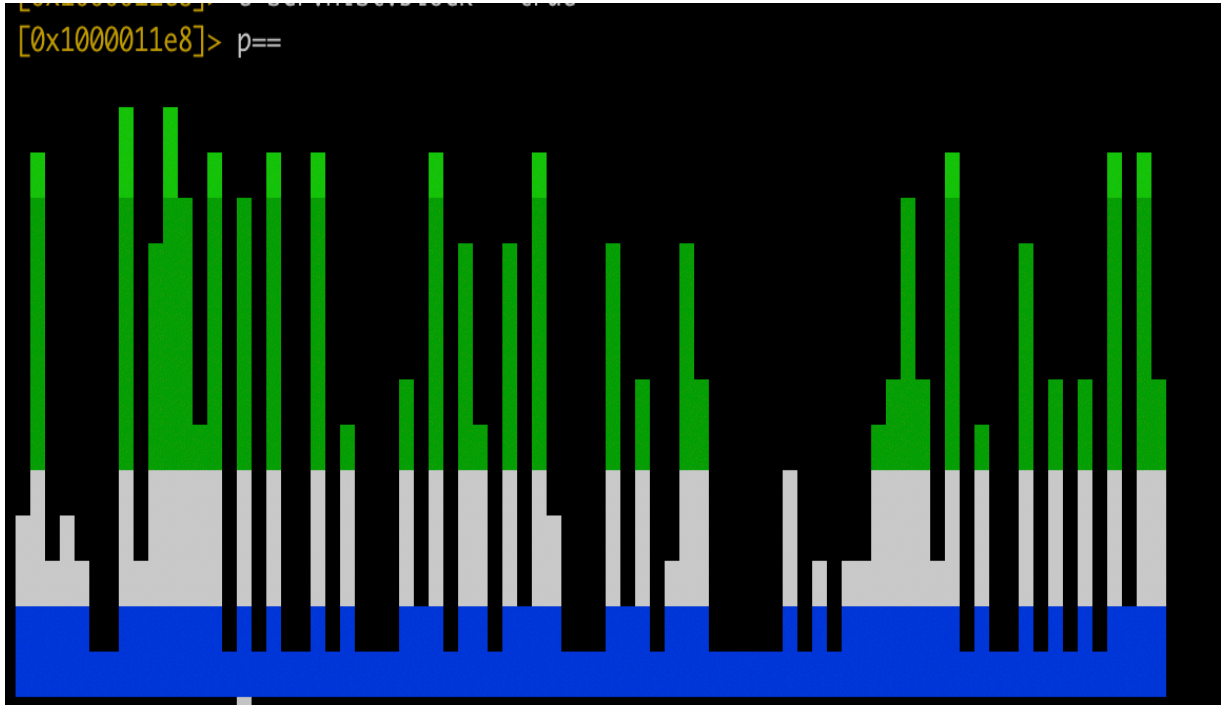
ASCII-ART COLOUR BARS

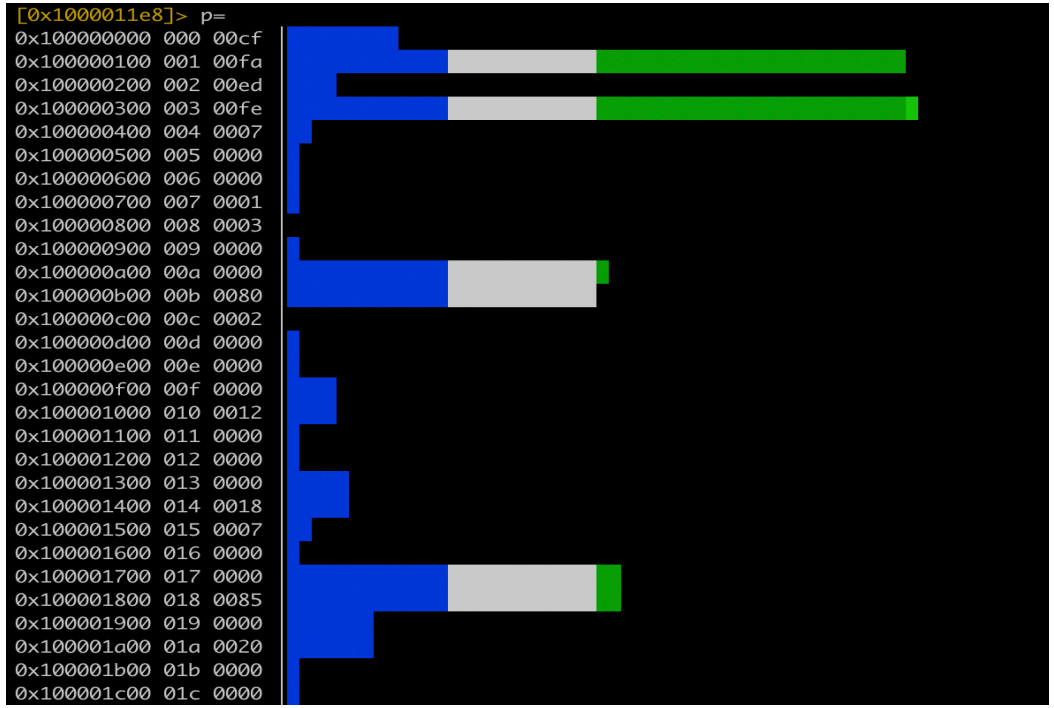


TASK 5: UTF-8 BASED IMPROVEMENTS:

- Added hist graphs support for p= and p==
- Can be enabled by setting *scr.hist.block = true*

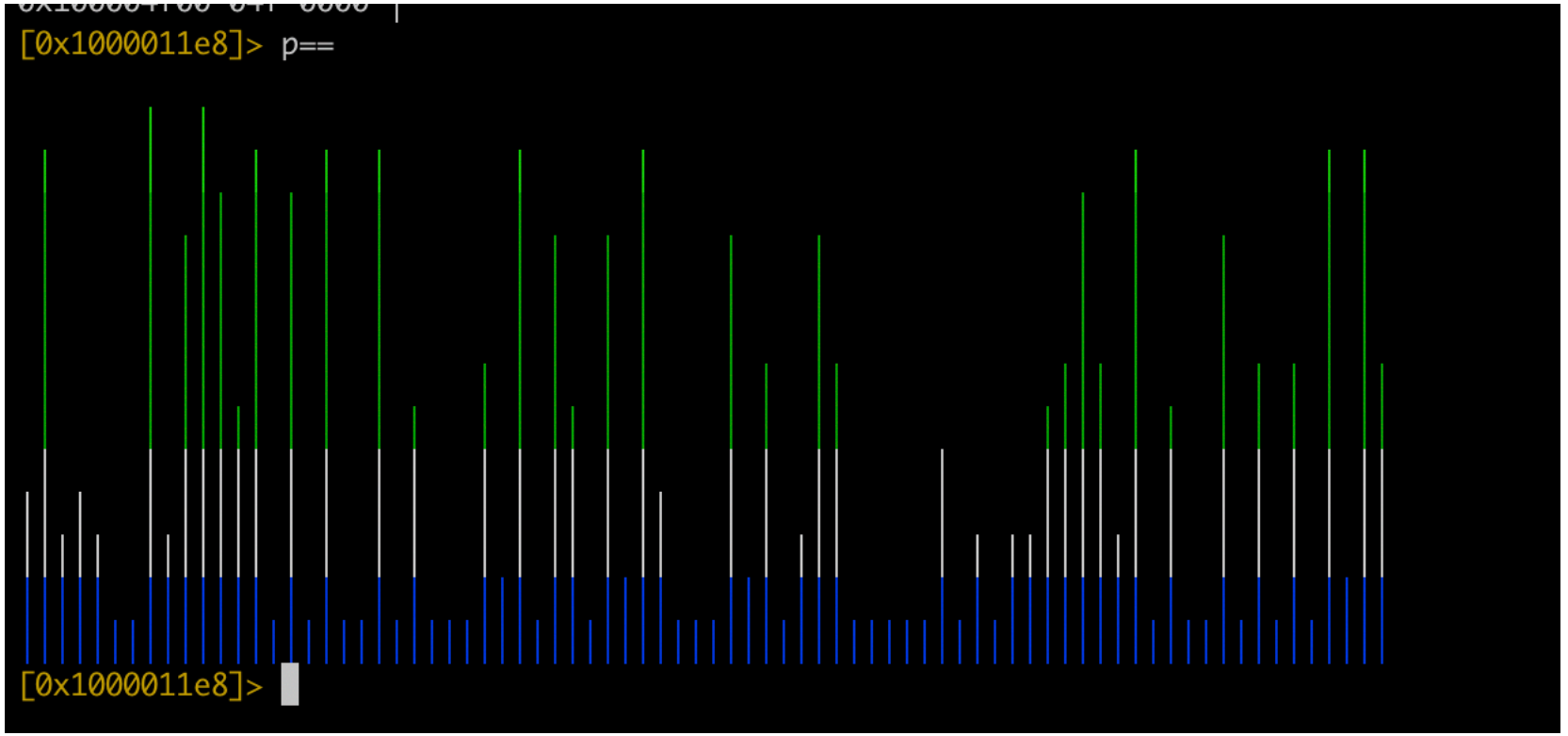
HIST GRAPHS


[0x1000011e8]> p==


[0x1000011e8]> p=
0x100000000 000 00cf
0x100000100 001 00fa
0x100000200 002 00ed
0x100000300 003 00fe
0x100000400 004 0007
0x100000500 005 0000
0x100000600 006 0000
0x100000700 007 0001
0x100000800 008 0003
0x100000900 009 0000
0x100000a00 00a 0000
0x100000b00 00b 0080
0x100000c00 00c 0002
0x100000d00 00d 0000
0x100000e00 00e 0000
0x100000f00 00f 0000
0x100001000 010 0012
0x100001100 011 0000
0x100001200 012 0000
0x100001300 013 0000
0x100001400 014 0018
0x100001500 015 0007
0x100001600 016 0000
0x100001700 017 0000
0x100001800 018 0085
0x100001900 019 0000
0x100001a00 01a 0020
0x100001b00 01b 0000
0x100001c00 01c 0000

TASK 5: UTF-8 BASED IMPROVEMENTS:

- Added the UTF-8 support to p= and p==, which are used when hist mode is disabled



TASK 6: TABLE API

- Due to time constraints, I was not able to work on this
- But pancake built it and it supports a lot of cool features like:
  - Special querying language to manipulate the table data
  - Exporting the graph data into CSV, SDB and other formats

TABLE API IN ACTION

```
[0x1000012a3]> p-h
```

offset	flags	funcs	cmts	syms	str
0x100000000	2	0	0	1	0
0x100000ee0	17	0	1	13	0
0x100001a08	1	0	0	1	0
0x100001dc0	2	0	0	2	0
0x100002530	1	0	0	1	0
0x1000028e8	2	0	0	2	0
0x100003058	2	0	0	2	0
0x100003410	5	0	0	5	0
0x1000037c8	1	0	0	1	0
0x100003b80	6	0	0	6	0
0x100003f38	2	0	0	2	0
0x1000042f0	79	0	2	77	0
0x1000046a8	6	0	1	0	5
0x100004a60	53	0	1	0	53
0x100004e18	86	0	4	0	24
0x1000051d0	28	0	3	0	0
0x100005588	1	0	1	0	0
0x100005cf8	1	0	0	0	0
	295	0	13	113	82

FUTURE WORK

- Add UTF-8 support for table API
- Integrate table API wherever necessary
- Build a tree API

ANY QUESTIONS?

THANK YOU