

Project 0

PintOS - Lists

In class, we discussed how to build a simple hello world test in PintOS, as well as some basic functionality of the list data structure included in the basic C Library present in PintOS. In this first project, your task is to implement another test to be run on top of Pintos and to practice with lists, as you may have to use lists in the next projects as well.

This project is not graded.

Sorting a List

- Change directory to `tests/threads` and create a file `list_sort.c`. Create a list of elements that have a field called `priority` (of type `int`).
- Create the function `populate` to fill list `l` with the elements of array `a`, which contains `n` integers.

```
void populate (struct list * l, int * a, int n)
```

- From iCorsi, download the file `listpop.h`, add it to `tests/threads` to populate your list using the array `ITEMARRAY`, which has `ITEMCOUNT` elements.
- Create the function `print_sorted` to sort the elements of list `l` and print them in ascending order of priority.

```
void print_sorted (struct list * l)
```

- Create a test function `test_list_sort()` and turn your code into a test to be run on top of Pintos, changing the necessary files (check the Hello World example).

Hints

- The functions above mentioned are the expected interface that you should implement as your solution. However, depending on your implementation, you could create other auxiliary functions as well.
- Use the functions declared in `list.h` to manipulate and sort your list.
- Don't forget that the allocation of list items is the user responsibility: `free()` all the memory space that you allocated with `malloc()`.

The output of a correct implementation when running the following command is shown in Figure 1:

```
$ pintos -v -- run name-of-the-test
```

Readings

- PintOS installation PDF (on iCorsi)
- PintOS documentation Chapter 1
 - Only Section 1.1

```
=====
                        Bochs x86 Emulator 2.6
                Built from SVN snapshot on September 2nd, 2012
                Compiled on Feb 17 2015 at 16:28:11
=====
00000000000i[      ] reading configuration from bochsrc.txt
00000000000e[      ] user_shortcut: old-style syntax detected
00000000000i[      ] installing nogui module as the Bochs GUI
00000000000i[      ] using log file bochsout.txt
Pilo hda1
Loading.....
Kernel command line: run test-list
Pintos booting with 4,096 kB RAM...
383 pages available in kernel pool.
383 pages available in user pool.
Calibrating timer... 204,600 loops/s.
Boot complete.
Executing 'test-list':
(test-list) begin
1 2 3 3 4 5 6 7 8 9
(test-list) end
Execution of 'test-list' complete.
```

Figure 1: The output of the list sorting test

Submission instructions

Please strictly follow the submission instructions defined here.

- Submit the following source files: `list_sort.c`.
- Complete and submit the report file renamed as: `ReportP0G<group_number>.pdf`.
- Compress everything into a single `.zip` archive named: `G<group_number>P0.zip`.
- Please upload only **one submission per group**: the resulting grade will apply to all members of the group.