

5-2-2013

# SoftDev

(Fall 2013)

First Name: \_\_\_\_\_

Last Name: \_\_\_\_\_

## INSTRUCTIONS

1. Write in clear, block capital letters
2. Don't use a red pen  
(pencils are OK, but then don't complain if "something is not what you wrote")
3. All the shell commands are intended to be issued on a BASH shell with the GNU tools
4. In questions with multiple choices, mark ALL the answers that are correct
5. If you don't know the answer... don't try to guess.  
A wrong answer in a multiple choice question has a NEGATIVE score
6. If you mark the wrong answer by mistake, write "NO" on its left side and I will ignore it

[A1] If you change the file permissions of a file, which file timestamps are updated?

- ☒ ctime  
☒ atime  
☐ mtime  
☐ none

[A2] Select which ones of the following commands prints only the 5<sup>th</sup> line of a file.

- ☐ cat file | tail -5 | head -1  
☐ cat file | sed '5p'  
☒ cat file | head -5 | tail -1 | sed '1q'  
☐ cat file | grep -v '^5'  
☐ cat file | sed '1,4d'  
☐ cat file | awk '{print \$5}'  
☒ cat file | awk '{if (NR==5) print \$0}'  
☐ cat file | awk 'NF~5 {print}'

[A3] If you need to locate a file in the filesystem, what is the main difference between the use of find and locate?

~~"find" will go through the file system / traverse folder structure at the moment of command~~

~~"locate" will use a database that has to be updated first~~

"find" WILL GO THROUGH THE FILE SYSTEM / TRAVERSE FOLDER STRUCTURE AT THE MOMENT OF COMMAND

"locate" WILL USE A DATABASE THAT HAS TO BE UPDATED FIRST



[A4] How can I combine sort and uniq to print the lines in fileB that are not in fileA?

```
sort fileB fileA | uniq -v
```

[A5] Given the following line of text:

one, 2, three, 4, "five", SIX

What is the text matched by the following regular expressions?  
(as if they were provided to grep -o for instance)

e.*\$	-> e, 2, three, 4, "five", SIX	e followed by arbitrary amount of chars, until end of line
(.)\1	-> ee	
[2-3]{2,3}	-> no match NO MATCH	

[A6] The file bb.txt contains the following text:

```
Elwood: It's ^106^ miles to Chicago,  
> we got a full tank of gas,  
> half a pack of cigarettes, it's dark...  
> and we're wearing sunglasses  
Jake: hit it.
```

What is the output of the following commands:

```
cat bb.txt | grep '[^>]'
```

```
awk '{print $1,$NF}' bb.txt
```

```
Elwood: Chicago,
```

```
> gas,
```

```
> dark.
```

```
> sunglasses
```

```
Jake: it.
```

```
sed '3s,s,d,' bb.txt
```

```
Elwood: It's ^106^ miles to Chicago,
```

```
> we got a full tank of gas,
```

```
> half a pack of cigarettes, it's dark.
```

```
> and we're wearing sunglasses
```

```
Jake: hit it
```

ONLY FIRST OCCURENCE  
NO GLOBAL FLAG

only first occurrence,  
global flag



[B1] Which of the following statements about function overloading in Python is true?

- ☐ two functions can have the same name, if the number of parameters is different
- ☐ two functions can have the same name and the same number of parameters, but it is not deterministic which one will be executed
- ☐ two functions can have the same name, if the parameters have different types
- ☒ two functions cannot have the same name

[B2] Mark the sentences that are true.

- ☐ Python dictionaries can use any object as keys
- ☒ In Python you can define a function inside another function
- ☒ Strings in Python are immutable objects
- ☐ .pyc is the extension of C libraries pre-compiled to be invoked from Python
- ☒ Using an uninitialized variable raises an exception
- ☒ Dictionary is a built-in type in Python

[B3] What is the output of the following pieces of Python code?

`print [x*(x-1) for x in (1,2,3,4)]`

[0, 2, 6, 12]

`print [['1','2','3','4'][x] for x in [1,0,3,2]]`

['2', '1', '4', '3']

`print [(x,y) for x in range(3) for y in range(3)]`

[(0,0), (0,1), (0,2), (1,0), (1,1), (1,2), (2,0), (2,1), (2,2)]

`l1 = [1,3,4,5]`

`l2 = l1`

`l2.append(6)`

`print l1`

[1, 3, 4, 5, 6]

[B4] You have a list like:

`L = [("2m", "5cm"), ("10m", "77cm"), ("1m", "82cm"), ("1m", "74cm")]`

in which each element is a tuple that represents the height of an object.

Complete the following line to properly order the list (from the smaller to the taller):

`L.sort(key= lambda x: 100*int(x[0][0:-2]) + int(x[1][0:-3]))`



[B5] Complete the output of this program...

```
def f(n):
    a, b = n, n
    while b <= a:
        yield a
        b = b * 2
        a = a + 10
```

```
print [x for x in f(4)]
```

a	b
4	4
14	8
24	16
34	32
44	64

[4, 14, 24, 34]

[C1] Mark the correct sentences

True False

- ☒ ☐ autotools is a suite of tools designed to simplify the portability of source packages in unix systems
- ☐ ☐ aclocal is used to automatically adjust the makefiles to the local system
- ☐ ☐ automake is a tool to generate the make script
- ☐ ☐ autoconf is a tool to generate the configure script
- ☒ ☐ configure is a shell script
- ☐ ☐ makefile.in is created by the configure script
- ☐ ☒ configure.ac macros are written in Lisp

[C2] Write three advantages of using dynamic libraries instead of static ones:

1. library can be used by multiple programs (lib only once in memory)
2. executable is smaller
3. plugin (no lib) can be updated without the need to know which application actually uses it (bugfix)

[C3] If hardware breakpoints are so convenient, why debuggers still implement software breakpoints as well?

- NOT ALL HARDWARE SUPPORTS HARDWARE BREAKPOINTS
- EVEN IF THEY ARE SUPPORTED, ONLY LIMITED NUMBER AVAILABLE

[C4] At which stage of the compilation process...

- the object files are combined together
- C macros are expanded
- syntax errors are detected
- the final executable is created
- the speed optimizations are applied
- the space optimizations are applied
- the required libraries are found and fetched
- C code is translated in assembler

- ☐ assembler ☐ preprocessor ☒ linker ☐ compiler
- ☐ assembler ☒ preprocessor ☐ linker ☐ compiler
- ☐ assembler ☐ preprocessor ☐ linker ☒ compiler
- ☐ assembler ☐ preprocessor ☒ linker ☐ compiler
- ☐ assembler ☐ preprocessor ☐ linker ☒ compiler
- ☐ assembler ☐ preprocessor ☐ linker ☒ compiler
- ☐ assembler ☐ preprocessor ☒ linker ☐ compiler
- ☐ assembler ☐ preprocessor ☐ linker ☒ compiler



[C5] Given the following simple makefile

```
CC = gcc
CFLAGS = -O1 -g -Wall
all: io.o data.o driver.o
    $(CC) $(CFLAGS) -o driver io.o driver.o data.o
data.o: data.c data.h
    $(CC) $(CFLAGS) -c data.c
driver.o: driver.c data.h
    $(CC) $(CFLAGS) -c driver.c
io.o: io.c
    $(CC) $(CFLAGS) -c io.c
```

and the project directory containing the following files:

-rwxr-xr-x	1	balzarot	balzarot	7765	2010-02-02	22:35	driver*
-rw-r--r--	1	balzarot	balzarot	252	2010-02-02	22:25	Makefile
-rw-r--r--	1	balzarot	balzarot	25	2010-02-02	22:14	data.c
-rw-r--r--	1	balzarot	balzarot	58	2010-02-02	22:34	driver.c
-rw-r--r--	1	balzarot	balzarot	25	2010-02-02	22:22	io.c
-rw-r--r--	1	balzarot	balzarot	9	2010-02-02	22:24	data.h
-rw-r--r--	1	balzarot	balzarot	1956	2010-02-02	22:30	data.o
-rw-r--r--	1	balzarot	balzarot	2232	2010-02-02	22:28	driver.o
-rw-r--r--	1	balzarot	balzarot	1948	2010-02-02	22:28	io.o

If, at the date "2010-02-02 23:21", the user gives the commands:

bash> make io.o

bash> make

Which files will be modified?

first cmd: nothing  
second cmd: driver.o, driver