

Announcements

Reminder: Quiz Friday; thereafter, [almost] every Wednesday

Women and Gender Minorities in Computing listserv sign-up (optional)

Office Hours canceled Friday this week

MathLAN issues

- Interactions between MathLAN and the college password/authentication system arose yesterday. Apparently, these issues were resolved last night, about 10:15 pm.
- System default settings apparently changed over the last year, causing much noise in a terminal window when emacs is started.
 - The noise is harmless (just annoying)
 - I hope to have instructions for muting the noise by Friday

Extra Credit Option:

- Thursday Extras: Thursdays@4:00 in Science 3821 (none this week)
- Friday, Sept. 6, 2pm, HSSC N3118: Marcus Eagan '12, Director of Developer Advocacy and Evangelism at Lucidworks: Open source projects, AI, Data Visualization, InfoSec, Search, Mobility, and robust APIs— sponsored by the Wilson Program
- Next week: Rosenfield Symposium on data, uses, power, etc.

Lab partners to be assigned over weekend for next week

- email me by Sunday noon if you prefer NOT to work with someone in the class this semester

CS Fall Picnic, Fri., Sept. 13 @ 5:00 in Merrill Park (West Shelter)

Linux and C Basics

Questions

Clicker Questions

Demo: Working on Labs, Window Layout

What is the difference between the following 2 Linux commands?

man directory

man -k directory

- A. These are just variations of same underlying command: "man" is a cryptic spelling of "manual", indicating an intention to search Linux's online documentation pages
- B. The second version has 3 more characters (including a space)
- C. The first version reports that no Linux command has the name "directory"
- D. The second version uses "-k" as a flag to indicate the manual should report all commands related to the keyword directory
- E. All of the above

What is the output of the following program?

```
#include <stdio.h>
```

```
int main()  
{  
    printf("hello");  
    printf("world!");  
    return 0;  
}
```

- A. hello
world!
- B. helloworld!
- C. hello world!
- D. Does not compile.
- E. None of the above.

What is the output of the following program?

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("I really "); /* Insert a descriptive
```

```
    printf("love ");      /* adjective here */
```

```
    printf("C.");
```

```
    return 0;
```

```
}
```

A. Doesn't compile.

B. I really C.

C. I really love C.

D. None of the above.

What is the output of the following program?

```
#include <stdio.h>
```

```
int main()
{
    int value;
    printf("value=%d\n", value);
    return 0;
}
```

- A. What's this %d business?
- B. value=0
- C. value=
- D. It's impossible to tell.
- E. I have no idea.

What is the result of the following program?

```
#include <eSpeakPackage.h>

int main() {
    eSpeakTalk("hi");
    return 0;
}
```

A.Doesn't compile.

B.Speaks "hi" in a male or female voice.

C.Speaks "hi" in a female or male voice and prints "hi" to terminal.

D.Generates an error when run.

E.None of the above.

When the following program is run, nothing is heard.

```
#include <eSpeakPackage.h>
int main() {
    eSpeakConnect( );
    eSpeakTalk("hi");
    eSpeakDisconnect( );
    return 0;
}
```

What might be wrong?

- A.Sound is muted
- B.Sound volume is too low to be heard
- C.Sound sent to headphones only, but headphones not used
- D.Any of the above
- E.Something else

Consider the following C program:

```
#include <stdio.h>
```

```
int main()  
{  
    /* Part A */  
    printf("one\n");  
    return 0;  
    /* Part B */  
    printf("two\n");  
    return 0;  
}
```

What happens when we try to compile and run this program?

- A. Output is
Part A
one
Part B
two
- B. Output is
Part A
one
- C. Output is
one
two
- D. Output is
one
- E. Compile or Run-time Error or something else

After compiling and running the previous program prog.c, we edit it to comment out the first return 0:

```
#include <stdio.h>
```

```
int main()
{
    /* Part A */
    printf("one\n");
    /* return 0;
    /* Part B */
    printf("two\n");
    return 0;
}
```

What happens when we try to compile and run this program as follows?

```
gcc -o prog prog.c
./prog    [or just prog]
```

- A. Compiler warning generated about comment inside another comment
- B. Compiler error
- C. No output generated
- D. Output is
one
- E. Output is
one
two

After running the original program prog.c, we create the following updated program

```
#include <stdio.h>
```

```
int main()  
{  
    /* Part A */  
    printf("one\n")  
    /* Part B */  
    printf("two\n");  
    return 0;  
}
```

What happens when we try to compile and run this program as follows?

```
gcc -o prog prog.c  
./prog    [or just prog]
```

- A. Compiler warning generated about comment inside another comment
- B. Compiler error
- C. No output generated
- D. Output is
one
- E. Output is
one
two