Spring 2022 courses

cc Quiz 3 - GCC, Org-mode, and Emacs

This is the third quiz of the term. It addresses the recent class contents (see recording/whiteboard in GDrive, or `notes.org` in GitHub). Feedback is provided where useful and applicable.

This quiz is not graded. You can play it once before the next class. You can play it unlimited times thereafter. The most difficult questions may reappear in the final exam.

Match the header argument and the meaning

`:main yes`	wrap `puts` statement in `int main(void) { }` function
`:includes stdio.h`	include standard input/output header file
`:tangle first.c`	export source code as C file `first.c` ("tangle")
`:exports both`	both result and source code will be exported
`:comments both`	link source code and org files, add comments to source
`:results raw`	insert output directly in org format into org file

Which of these are valid `PATH` settings?

TRUE:

- `C:\Program Files\Emacs\x86_64\bin`
- `C:\Program Files (x86)\MinGW\x64\bin`

FALSE:

- `C:/Program Files/Emacs/x86_64/bin`
- `C:/Program Files (x86)/MinGW/x64/bin`
- `c:/usr/bin`

Feedback: Under Windows, the folders in the search `PATH` are separated by a backslash (`\`), while on MacOS/Linux, they are separated by a forward slash (`/`). Under MacOS/Linux, path names have no drive identifier like `E:` or `C:`.

The `PATH` is a variable only for the Windows operating system

FALSE

Feedback: Every computer must search for a program when you ask for it. The `PATH` variable is the list of folders that the computers searches, in Windows, MacOS/Unix and Linux.

Know your shell

The expressions `--version` and `-nw` in the following code block are program options or flags.

```
prompt> gcc --version
prompt> emacs -nw
```

TRUE

Feedback: `gcc --version` returns the program version that is installed on your PC, while `emacs -nw` opens Emacs in the terminal (without certain graphics capabilities), as long as either program is in the `PATH`.

You can configure GNU Emacs using a "hidden" initial configuration file `~/.emacs`

TRUE

Feedback: the dot (`.`) before the file name marks this file as "hidden" in the sense that you normally need to tell the Operating System that you wish to see these files, too (except under Linux, of course, where users are not afraid of extra information).

Which symbols mark Org-mode metadata?

Tip: Org-mode meta data include the document title, author information, and different options for how to render the Org-mode file.

TRUE:

- `#+`

FALSE:

- `##`
- `/* */`
- `//`

Feedback: Examples are `#+CAPTION:` for an image caption, `#+TITLE:`, `#+OPTION:` and `#+STARTUP:`. Check any of my Org-mode files on GitHub to see these in action. `##` is Org-mode comment, `/* */` and `//` are C programming comments.

Font, font size and theme of a GNU Emacs buffer cannot be changed

FALSE

Feedback: To change the font size, use `C-x C-=`. To change the custom theme, enter `M-x custom-themes RET`. For font setting information, check the Emacs manual, or type `C-h a font RET` (`a~propos ~font`).

Keybindings for Emacs functions can be changed

Tip: a keybinding means that a function - e.g. save-file - can be bound to different keyboard key sequence.

TRUE

Feedback: Emacs is fully customizable. You can, for example, switch Emacs' kill/yank keys (`C-w~/~C-y`) to the keys for cut/copy/paste that you're used to, by entering `M-x cua-mode RET` (toggles - i.e. applying the function again returns to the default state).

How can you abort or quit any command in GNU Emacs?

TRUE:

- `C-g`

FALSE:

- `quit`

Feedback: `q()` or `quit()` exits the R shell program, not Emacs.

How can you see all Emacs buffers that are currently open?

TRUE:

- `M-x list-buffers`
- `C-x C-b`

FALSE:

- `ListBuffers()`
- `C-M-\`

Feedback: Each key sequence, like `C-x C-b`, corresponds to a Lisp function, in this case `list-buffers`. To find the function belonging to a key, type `C-h k <key sequence>`. `C-M-\` is the `indent-region` function. Check it yourself!

How can you close all buffers except one?

TRUE:

- `C-x 1`

FALSE:

- `C-x 2`
- `C-x 3`
- `C-x 5 2`
- `C-x 5 0`

Feedback: `C-x 2~/~C-x 3` opens another horizontal/vertical buffer window. `C-x 5 2` creates another Emacs frame. `C-x 5 0` closes the current frame.

cc Quiz 2 - C vs C++ and infrastructure

Update notice

> Quiz 2 (10 questions) is now available to you. Try to complete it before class tomorrow (11 AM). Takes no more than 5-10 minutes. Cheers!

Settings text

> This is the second quiz of the term. It addresses last week's issues - especially the last part of the introduction to C (C vs. C++, and the installation of a C compiler). Feedback is provided where

useful and applicable.

This quiz is not graded. You can play it once before the next class. You can play it unlimited times thereafter. The most difficult questions may reappear in the final exam.

What's the difference between C and C++?

Match the property and the language.

C	C++
Created by Thompson/Ritchie 1970s	Created by Stroustrup 1980s
Imperative procedural langauge	Object-oriented Programming language
System programming language	More used for Games programming
Important for Internet of Things (embedded computing)	Important for airflight Software

In OOP, classes are used for code reuse and maintenance

OOP stands for Object-Oriented Programming, a programming paradigm.

TRUE

Feedback: An example is the PowerPlant class that has the subclasses NuclearPowerPlant, CoalPowerPlant, and GasPowerPlant. Code for the PowerPlant class can be reused in the subclasses (so-called "inheritance").

C is an object-oriented programming language

FALSE

Feedback: C++ is an object-oriented superset of C. C is not OOP, but C++ is. Other OOP languages are: Java, C#, Smalltalk.

What do you need to create and run a C program?

TRUE:

- An editor (like Emacs)
- An operating system (like Windows)
- A compiler + linker (like GCC)

FALSE:

- A hosting platform (like GitHub)
- A learning management system (like Schoology)

Know your code

Match the term and the definition!

Source code	Human-readable program

Machine code	Program ready for execution
Object code	Code produced by a compiler

Which of these are valid `PATH` settings?

TRUE:

- `C:\Program Files\Emacs\x86_64\bin` (in Windows)
- `C:\Program Files (x86)\mingw-w64\bin` (in Windows)
- `/usr/bin` (in MacOS/Linux)

FALSE:

- `C:/Program Files/Emacs/x86_64/bin` (in Windows)
- `C:/Program Files (x86)/mingw-w64/bin` (in Windows)
- `E:\usr\bin` (in MacOS/Linux)
- `E:/usr/local/bin` (in MacOS/Linux)

Feedback: Under Windows, the folders in the search `PATH` are separated by a backslash (`\`), while on MacOS/Linux, they are separated by a forward slash (`/`). Under MacOS/Linux, path names have no drive identifier like `E:` or `C:`.

The `PATH` tells the PC where to look for a program

TRUE

Know your shell

The `--version` (in Windows) or `-v` (in MacOS) is a program option (or flag).

```
Windows> gcc --version
MacOS>   cc -v
```

TRUE

Feedback: This option for the programs `gcc` or `cc` returns the program version that is installed on your PC.

What is GCC?

We installed GCC on Windows as MinGW.

TRUE:

- Open source software and part of the GNU project
- A bundle of programs including the `gcc` C compiler

FALSE:

- A standard library for C programming
- A program to translate machine code into source code

Feedback: GCC is the GNU Compiler collection, which allows compilation (transformation of source code into machine or object code) for a bunch of different languages including C and C++. `gcc` is one of these compiler programs. A standard library (like `stdio.h`) is common for most high level languages (like C and C++). Alas, under Windows, GCC and gcc are not distinguished (try it).

The "binary" of a program (like `gcc`) only runs on your type of PC

TRUE

Feedback: a binary is machine code built from source code, specifically built for your PC's architecture, including the specific chips that make up your CPU.

cc Quiz 1 - introduction to C and to the course

This is the first weekly quiz. Time = 15 minutes, 10 points max.

GNU Emacs is a literate programming tool

TRUE

> GNU Emacs is a customizable, extensible editor. Via Org-mode, a major environmental mode, Emacs can be used to write programs in a "literate" style, by mixing documentation, code and output.

Where does C come from?

TRUE:

- C is a by-product of the UNIX operating system | C was written as a language for Unix, and then Unix was rewritten in C.

FALSE:

- C is a by-product of the Linux operating system | UNIX, and C are 30 years older than Linux.
- C was the first programming language | Not by a long shot. When C was created, there were already hundreds of programming languages in use.

Which of the following are strengths of the C programming language?

TRUE:

- Portability | C programs can be compiled and executed on almost any computer system
- Standard Library | Because it is so old and established, many functions for C are prewritten and bundled in libraries that you can just use
- Speed and efficiency | It's fast and small programs can achieve a lot

FALSE

- Permissiveness | This refers to the fact that it is easy to make mistakes in C - the language doesnt check itself very carefully
- Terseness and understanding for humans | C is very terse, the source code is hard to read because it is highly compressed
- Maintenance of large programs | C is harder to maintain for large programs

Android (for phones) is largely written in Python

FALSE

> Android is essentially Linux. Operating systems (OS) are usually written (mostly) in C. Python is much too slow a language for OS needs.

How will this course be graded?

TRUE:

- Class assignments: 10%
- Monthly tests: 30%
- Lab projects: 30%
- Final exam: 30%

Match glossary items

Note: all terms relate to the use of the word in computer science

TERM	DEFINITION
Software Library	Bundle of useful functions
Portability	Ability of software to run on different hardwares
Algorithms	Fixed process or set of rules
String	A data type representing text
UNIX	Operating System (ca. 1969)
Compiler	Software to translate source code into machine code

Match GitHub action and definition

ACTION	MEANING
Create repository	Create a place for your code
Create branch	Create a modified copy of a main repository
Open pull request	Request for changes of main repository
Merge branch	After review, merge changes into main repo
Submit issue	Make a comment on a repository
Commit	Make changes permanent

What's a computer?

Which of these makes up a computer?

TRUE:

- CPU + Memory + Harddisk | You don't need peripherals, though they are nice to have
- x86-64 + 4GB RAM + 500GB SSD | Chip set + volatile memory + non-volatile memory

FALSE:

- CPU + Keyboard + Mouse | Memory is missing (RAM and Hard disk)
- Editor + Screen + Motherboard | Editor is an application only, screen and keyboard are peripherals - CPU and memory are missing

How are programs processed?

Programming consists of a series of steps. Bring them in the right order, beginning with the creation of source code.

1. write program in editor (harddisk)
2. compile program to machine format (RAM)
3. run program (CPU)
4. display results (harddisk)

Literate programs contain documentation + code + output

TRUE

> There is more to literate programming, but this is what you should remember. Programming in this way, e.g. by using interactive notebooks, directly helps you develop your computer and data literacy.

Author: Marcus Birkenkrahe
Created: 2022-03-27 Sun 16:44
Validate