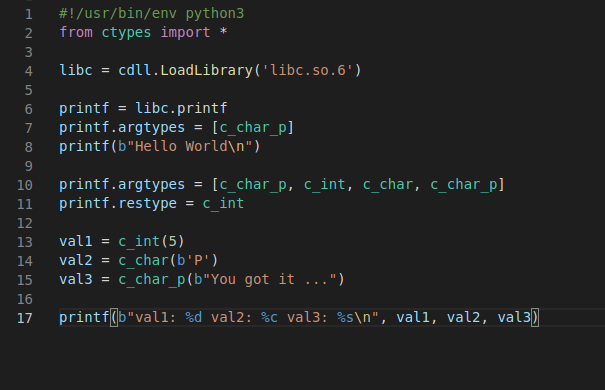
Question 1:

1. It prints out the string provided, the \ next to the “” make it so it doesn’t recognize them at delimiters for the string but instead as characters to print out
2. It signifies it as a byte string so it prints it off the whole string
3. Argtypes specifies which C data types to import from the library, restype sets what return type is expected (c\_int by default)
4. It is the return value of line 15 in c\_int format
5. Windll.LoadLibrary(‘kernal32’)

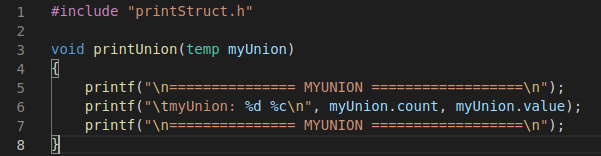
Question 2:

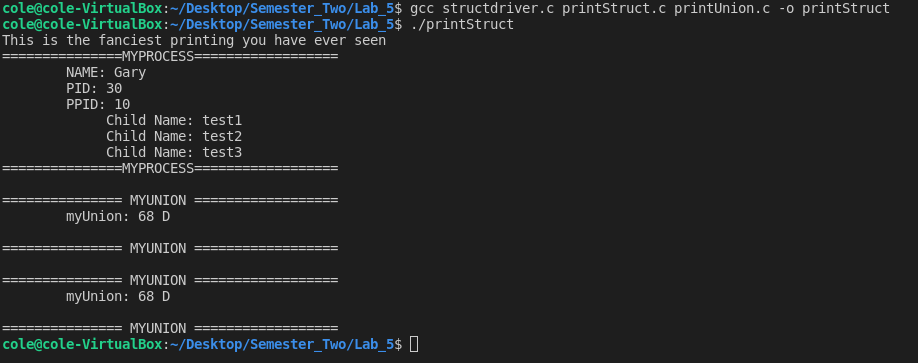
1. It is a symbolic link to the 64-bit x86 C library
2. /usr/lib/x86\_64-linux-gnu/libc.so.6
3. c\_char\_p is the pointer to the beginning of the string, c\_int is %d and c\_char is %c
4. 
5. printf.argtypes = [c\_char\_p, c\_int, c\_char, c\_char\_p]

Question 3:

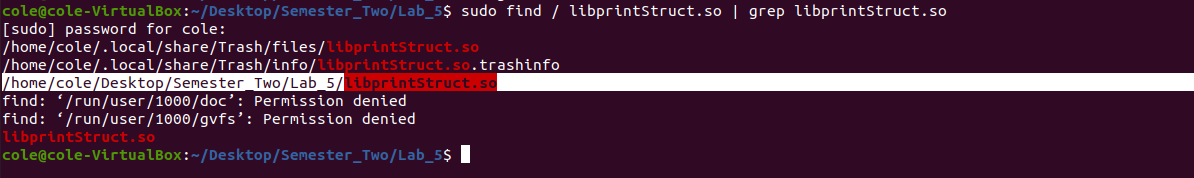
1. to define process struct and temp union and to provide the prototype for printStruct function
2. to define the printStruct function
3. to use the printStruct function and the structs defined in printStruct.h

a. yes, provided you create a new printUnion function

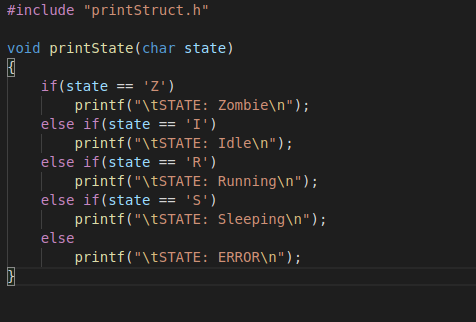
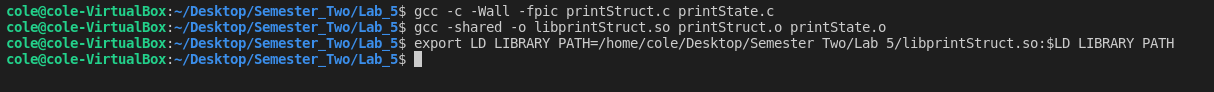
b. 

c. 

Question 4:

1. – shared : Produce a shared object which can then be linked with other objects to form an executable. -c : Compile or assemble the source files, but do not link. -Wall : enables all the warnings about constructions. -fpic : Generate position-independent code (PIC) suitable for use in a shared library, if supported for the target machine.
2. In the directory where we created it: 
3. To link it globally for use in other programs, not just ones in the same directory

Question 5:

1. 10 – ppid, 11 – pid, b’Process’ – name, 0x7A – both count AND value (count is int and value is char of 0x7A)
2. 1. Add a state char to the definition in printStruct.h 2. Add a state c\_char to the definition in printStruct.py 3. Add a line to print out the value in printStruct.c .
3. 
4. Yes, using the following two lines:

libstruct.printState.argtypes = [c\_char]

libstruct.printState(b'Z')

1. You can change the parameters easily using argtypes