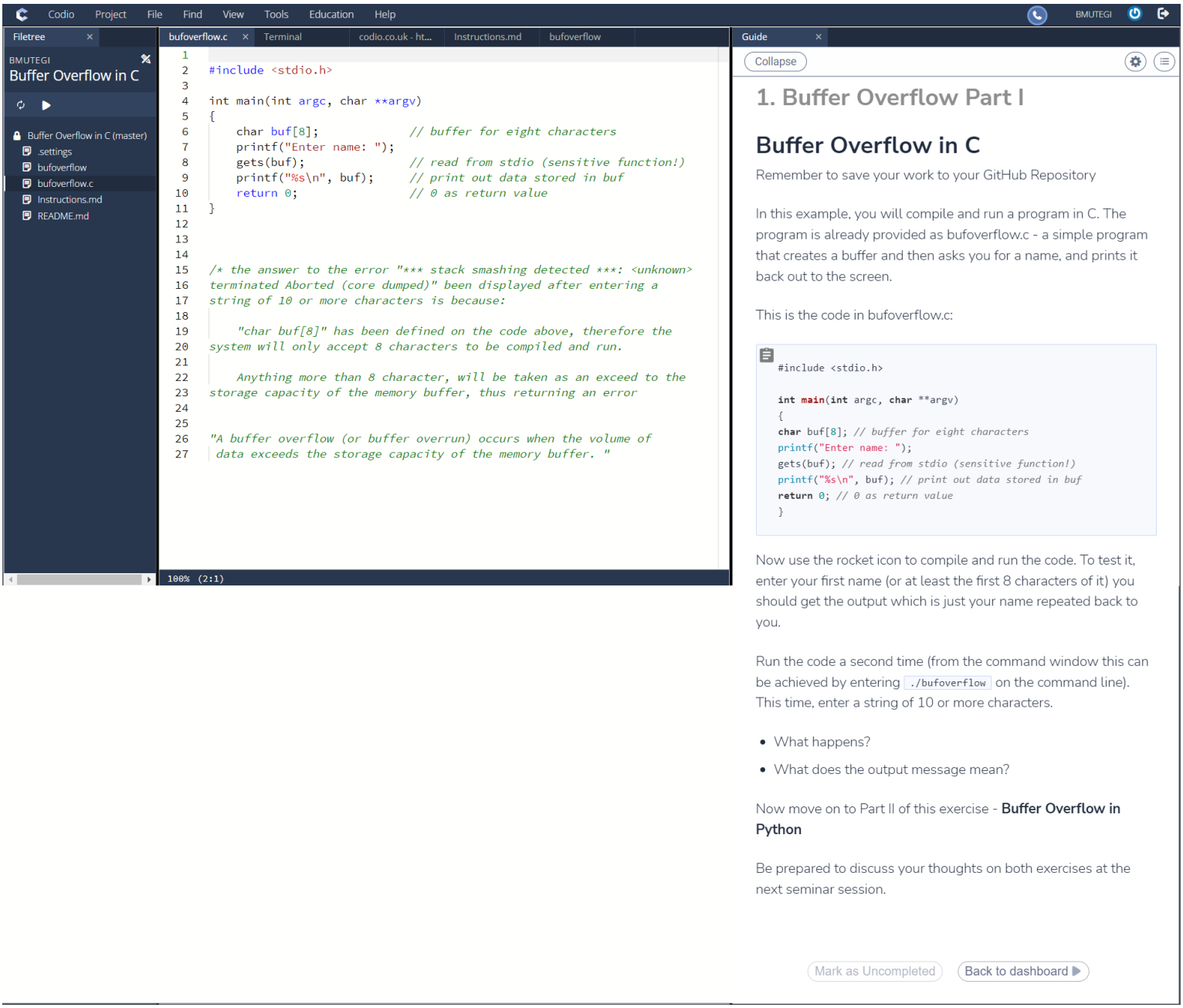
Codio
Project
File
Find
View
Tools
Education
Help
BMUTEGI
Filetree
BMUTEGI
Buffer Overflow in C
Buffer Overflow in C (master)
.settings
bufoverflow
bufoverflow.c
Instructions.md
README.md
bufoverflow.c
Terminal
codio.co.uk - ht...
Instructions.md
bufoverflow
#include <stdio.h>
int main(int argc, char **argv)
{
char buf[8]; // buffer for eight characters
printf("Enter name: ");
gets(buf); // read from stdio (sensitive function!)
printf("%s\n", buf); // print out data stored in buf
return 0; // 0 as return value
}
/* the answer to the error "*** stack smashing detected ***: <unknown>
terminated Aborted (core dumped)" been displayed after entering a
string of 10 or more characters is because:
"char buf[8]" has been defined on the code above, therefore the
system will only accept 8 characters to be compiled and run.
Anything more than 8 character, will be taken as an exceed to the
storage capacity of the memory buffer, thus returning an error
"A buffer overflow (or buffer overrun) occurs when the volume of
data exceeds the storage capacity of the memory buffer. "
100% (2:1)
Guide
Collapse
1. Buffer Overflow Part I
Buffer Overflow in C
Remember to save your work to your GitHub Repository
In this example, you will compile and run a program in C. The program is already provided as bufoverflow.c - a simple program that creates a buffer and then asks you for a name, and prints it back out to the screen.
This is the code in bufoverflow.c:
#include <stdio.h>
int main(int argc, char **argv)
{
char buf[8]; // buffer for eight characters
printf("Enter name: ");
gets(buf); // read from stdio (sensitive function!)
printf("%s\n", buf); // print out data stored in buf
return 0; // 0 as return value
}
Now use the rocket icon to compile and run the code. To test it, enter your first name (or at least the first 8 characters of it) you should get the output which is just your name repeated back to you.
Run the code a second time (from the command window this can be achieved by entering ./bufoverflow on the command line). This time, enter a string of 10 or more characters.
What happens?
What does the output message mean?
Now move on to Part II of this exercise - Buffer Overflow in Python
Be prepared to discuss your thoughts on both exercises at the next seminar session.
Mark as Uncompleted
Back to dashboard