CSc 3320: Systems Programming

Spring 2021

Homework

# 3: Total points 100

Submission instructions:

1. Create a Google doc for each homework assignment submission.
2. Start your responses from page 2 of the document and copy these instructions on page 1.
3. Fill in your name, campus ID and panther # in the fields provided. If this information is missing in your document TWO POINTS WILL BE DEDUCTED per submission.
4. Keep this page 1 intact on all your submissions. If this *submissions instructions* page is missing in your submission TWO POINTS WILL BE DEDUCTED per submission.
5. Each homework will typically have 2-3 PARTS, where each PART focuses on specific topic(s).
6. Start your responses to each PART on a new page.
7. If you are being asked to write code copy the code into a separate txt file and submit that as well.
8. If you are being asked to test code or run specific commands or scripts, provide the evidence of your outputs through a screenshot and copy the same into the document.
9. Upon completion, download a .PDF version of the document and submit the same.

Full Name: Venkata Mani Mohana Rishitha Srikakulapu

Campus ID: vsrikakulapu1

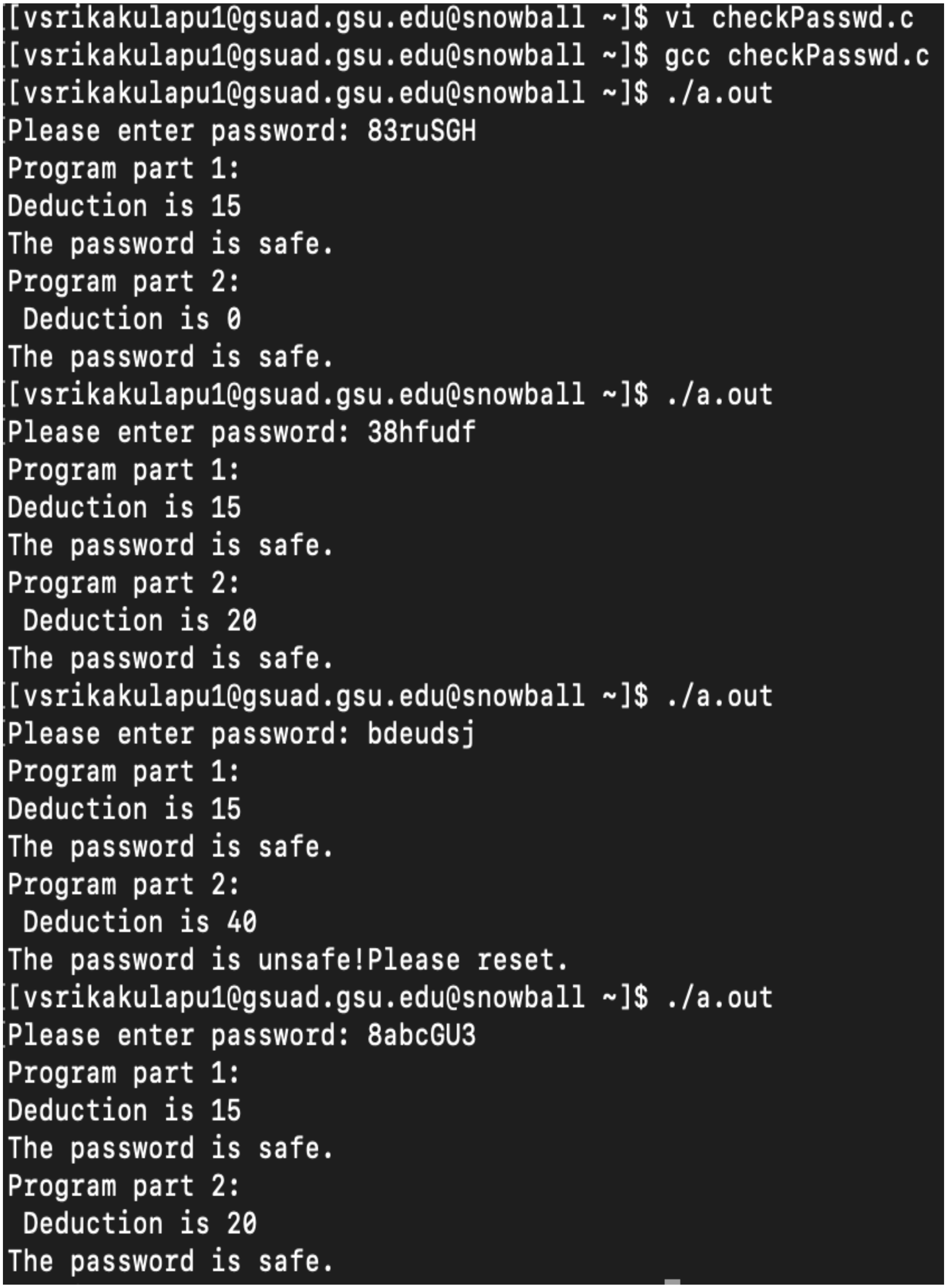
Panther #: 002523638

PART 1: 40pts

1. Write a C program checkPasswd.c to check if the length of a given password string is 10 characters or not. If not, deduct 5 points per missing character. If the total deduction is greater than 30 points, print out the deduction and message "The password is unsafe! Please reset."; otherwise, print out "The password is safe."

2. Similar to above question, update the C program checkPasswd.c to check if a password is safe or by not by checking only the evaluation criteria below. It will still print out the final score, and "safe" or "unsafe" when deduction is more than 30 points.

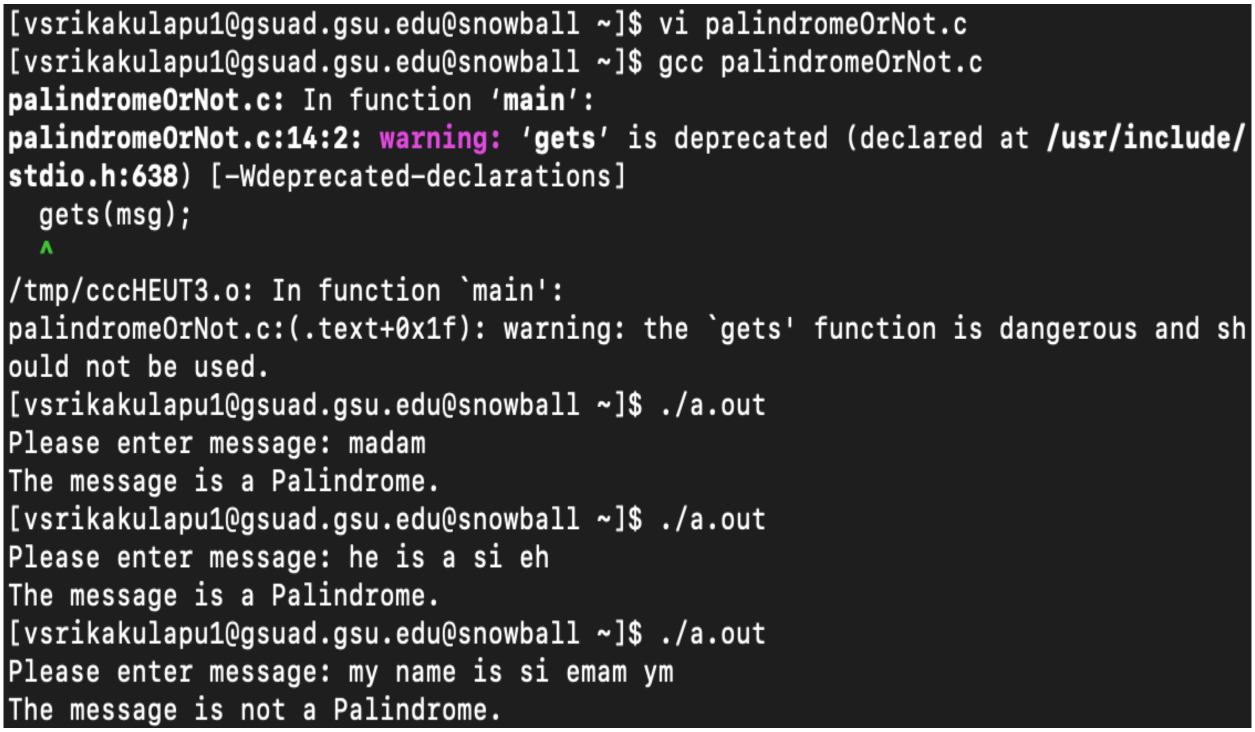
- Missing lower case -20 points
- Lack of capital letters -20 points
- Missing numbers -20 points
- More than 2 consecutive characters (e.g. 123 or abc) -20 points



```
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ vi checkPasswd.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ gcc checkPasswd.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter password: 83ruSGH
Program part 1:
Deduction is 15
The password is safe.
Program part 2:
 Deduction is 0
The password is safe.
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter password: 38hfudf
Program part 1:
Deduction is 15
The password is safe.
Program part 2:
 Deduction is 20
The password is safe.
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter password: bdeudsj
Program part 1:
Deduction is 15
The password is safe.
Program part 2:
 Deduction is 40
The password is unsafe!Please reset.
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter password: 8abcGU3
Program part 1:
Deduction is 15
The password is safe.
Program part 2:
 Deduction is 20
The password is safe.
```

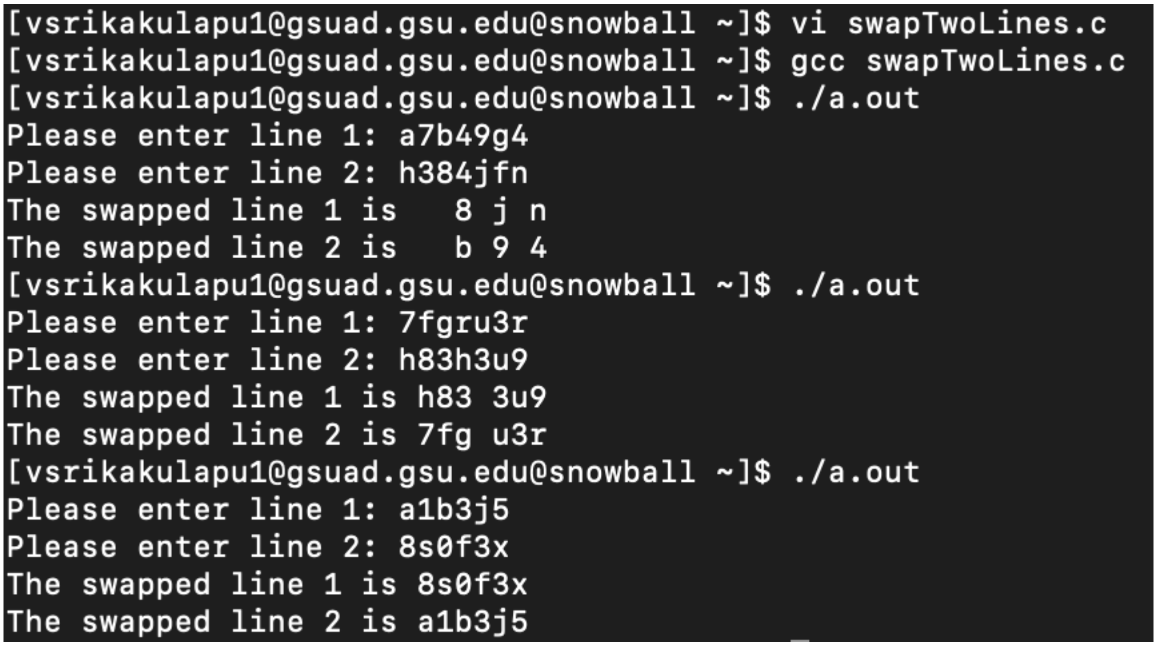
Part II : 40pts

3. Write a program that reads a message (can be characters, numeric or alphanumeric) and checks whether it is a palindrome (the characters in the message are the same when read from left-to-right or right-to-left).



```
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ vi palindromeOrNot.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ gcc palindromeOrNot.c
palindromeOrNot.c: In function ‘main’:
palindromeOrNot.c:14:2: warning: ‘gets’ is deprecated (declared at /usr/include/
stdio.h:638) [-Wdeprecated-declarations]
  gets(msg);
  ^
/tmp/cccHEUT3.o: In function `main':
palindromeOrNot.c:(.text+0x1f): warning: the `gets' function is dangerous and sh
ould not be used.
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter message: madam
The message is a Palindrome.
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter message: he is a si eh
The message is a Palindrome.
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter message: my name is si emam ym
The message is not a Palindrome.
```

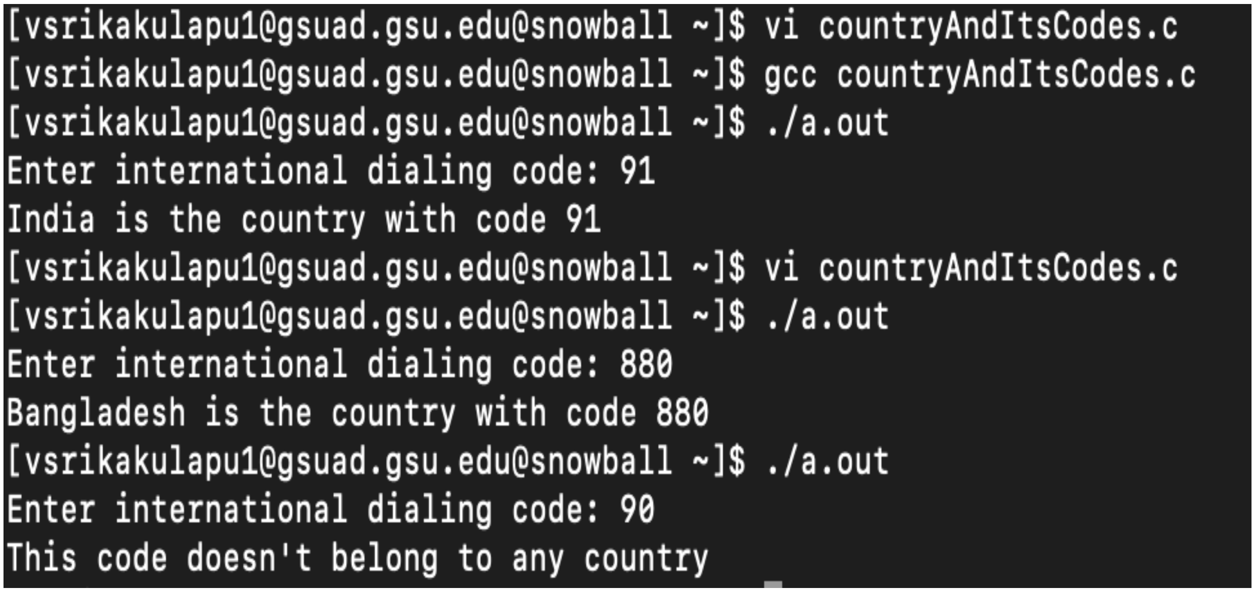
4. Write a program that will swap two variables without the use of any third variable. Utilize this program to write a program that reads two sentences that contain alphanumeric characters and the program must swap all the numerics in sentence1 with alphabet characters from sentence 2 and vice-versa. Keep the lengths of the sentences as identical.



```
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ vi swapTwoLines.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ gcc swapTwoLines.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter line 1: a7b49g4
Please enter line 2: h384jfn
The swapped line 1 is   8 j n
The swapped line 2 is   b 9 4
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter line 1: 7fgru3r
Please enter line 2: h83h3u9
The swapped line 1 is h83 3u9
The swapped line 2 is 7fg u3r
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Please enter line 1: a1b3j5
Please enter line 2: 8s0f3x
The swapped line 1 is 8s0f3x
The swapped line 2 is a1b3j5
```

Part III : 20pts

5. Write a program that asks the user to enter an international dialing code and then looks it up in the country_codes array (see Sec 16.3 in C textbook). If it finds the code, the program should display the name of the corresponding country; if not, the program should print an error message. For demonstration purposes have at least 20 countries in your list.
(Programming Project 1 on pg412 in C textbook)



```
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ vi countryAndItsCodes.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ gcc countryAndItsCodes.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Enter international dialing code: 91
India is the country with code 91
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ vi countryAndItsCodes.c
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Enter international dialing code: 880
Bangladesh is the country with code 880
[vsrikakulapu1@gsuad.gsu.edu@snowball ~]$ ./a.out
Enter international dialing code: 90
This code doesn't belong to any country
```