CSc 3320: Systems Programming

Spring 2021

Homework

4: 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 *submission's 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: Mahi Berhanu

Campus ID: mberhanu2

Panther #: 002478520

Part I:

1.

```
//Mahi Berhanu

//check if password length and output score and whether it is
safe or not

#include <stdio.h>

#include <string.h>

int main() {

 char password[10]; //declare password

 printf("Please Enter a Password: \n");//ask for user input

 scanf("%s", password);

 int length; // declare all variables needed

 int score;

 int extra_length;

 length = strlen(password);



 if(length != 10 ){ //if statement to check if the length is
not equal to 10

   extra_length = 10 - length;

   score = 5 * extra_length;

  if(score == 0){

     printf("Score is: %d; the password is safe", score);

   } else if(score >= 30 ){// print statement
```

```
     printf("Score is: %d; the password is unsafe! Please
reset. \n", score);

   } else

     printf("Score is: %d; the password is safe", score);

 return 0;

}

}
```

```
Please Enter a Password:
mahi1234
Score: 10
 The password is safe>
```

2.

```
//Mahi Berhanu

// Check if password fulfils criteria and print out score

#include <stdio.h>

#include <string.h>

int main() {

 char password[100]; //declare password

 printf("Please Enter a Password: \n");//ask for user input

 scanf("%s", password);

 int length = strlen(password);// declare all variables needed

 int score;
```

```
 int lower_case = 0; int upper_case = 0;int number = 0; int
consecutive = 0;

 // check if password has upper case letters if it does
increment

 for (int i = 0; i < length; i++){

   if(password[i] >= 'A' && password[i] <= 'Z'){

     upper_case ++;

   }

 }

 // if no upper case take points off

  if(upper_case <=0){

    score -= 20;

  }

  // check if password has lower case letters if it does
increment

  for (int i = 0; i < length; i++){

   if(password[i] >= 'a' && password[i] <= 'z'){

     lower_case ++;

   }

  }

  // if no lower case take points off

  if(lower_case <=0){

    score -= 20;

  }
```

```
  // check if password has consecutive  letters if it does
increment

  for (int i = 0; i < length; i++){

    for(int j =i+1; j<length; j++){

      if(password[j] - password[i] == 1){

        consecutive++;

      }

    }

  }

  // if more than 2 letters consecutive take points off

    if(consecutive > 2){

    score -= 20;

   }

   // check if password has numbers if it does increment

   for (int i = 0; i < length; i++){

     if(password[i] >= '0' && password[i] <= '9'){

     number ++;

     }

   }

   // if no numbers take points off

   if(number <=0){

    score -= 20;

   }
```

```
   if(score == 0){

     printf("Score is: %d; the password is safe", score);

   } else if(score <= 30 ){// print statement

     printf("Score is: %d; the password is unsafe! Please
reset. \n", score);

   } else

     printf("Score is: %d; the password is safe", score);

 return 0;

}
```

```
Please Enter a Password:
Mahi
Score is: -20; the password is unsafe! Please reset.
>
```

Part II:

3.

```
// Mahi Berhanu

//Checks if a word is a palindrome

#include <stdio.h>

#include <string.h>

#include <ctype.h>


void isPali(char* str){

 //initializing variables

 char * left = str; // left is equal to user input

  int length = strlen(str);

 char *right = left + length - 1;

 while (*left != '\0'){ //if left isn't null increment

   left++;

 }

 int i;

 // ignore special characters and space

while (('a' <= str[i] && 'z' >= str[i]) &&

    !('A' <= str[i] && 'Z' >= str[i]) &&

    !('0' <= str[i] && '9' >= str[i])){

 for(left = str; right >= left;){ // loop
```

```
   if(*right == *left){
     right --;
     left ++;
   }
   else
   break;
 }
 break;
}
//print out statement
 if(left >right){
     printf("The word is Palindrome");
 }else
 printf("The word is not Palindrome");
}
//driver code
int main() {
char str[100];//assigns size for word
printf("Enter a word: ");
gets(str);// gets the input of the user
//print out statement
isPali(str);
```

```
 return 0;

}
```

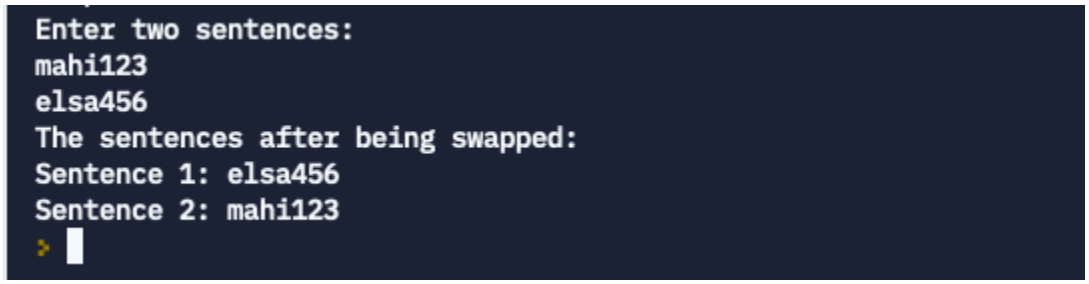
```
Enter a word: madam
The word is Palindrome>
```

4.

```
//Mahi Berhanu

//Swap two variables without any additional variable

#include <stdio.h>

//function that will swap the two characters

void swap(char *x, char *y)

{

*x=*x+*y;

*y=*x-*y;

*x=*x-*y;

}


int main()

{

//declare the characters
```

```
char s1[200];

char s2[200];

int i=0;

 //ask for user input and scans

printf("Enter two sentences:\n");

scanf("%[^\n]%*c", s1);

scanf("%[^\n]%*c", s2);

// check the characters while s1 isn't empty and swaps the
digits from s1 to s2 and the characters as well

while(s1[i]!='\0')

{

if(isalpha(s2[i]))

{

swap(&s1[i], &s2[i]);

}

else if(isalpha(s2[i]))

{

swap(&s1[i], &s1[i]);

}else if(isdigit(s2[i]))

{

swap(&s1[i], &s2[i]);

}

else if(isdigit(s2[i]))
```

```
{
swap(&s1[i], &s1[i]);
}
i++;
}
//print out statement.
printf("The sentences after being swapped: \n");
printf("Sentence 1: ");
puts(s1);
printf("Sentence 2: ");
puts(s2);
return 0;
}
```



Part III:

5.

```
 //Mahi Berhanu

//Match user input with a country area code

#include <stdio.h>

int main() { //decare struct to store country and dialing code

 struct dialing_code{

 char *country;

 int code;

};

//declare country codes

const struct dialing_code country_codes[] = {

 {"Argentina",  54}, {"Brazil", 55}, {"China", 86},

 {"Ethiopia", 251}, {"United States", 1},{"France", 33},

 {"India",  91}, {"Egypt", 20}, {"Japan",  81}, {"Iran", 98},

 {"Mexico", 52}, {"Italy", 39}, {"Sudan",  249}, {"Turkey",
90},

 {"South Africa", 27}, {"Poland", 48}, {"Pakistan", 92},

 {"Thailand", 66}, {"Spain", 34}, {"Nigeria", 234}};

 //initialize variables

 int code;

 int found = 0;

 //Ask for user input

 printf("Please Enter an International Dialing Code: \n");
```

```
 scanf("%d", &code);

 // loop as i < the length of array and check if there is any
match b/n user and country codes we have

 for(int i=0; i<21; i++){

   if(country_codes[i].code == code){

     printf("The Country that matches the area code you
entered is: %s\n", country_codes[i].country);// if there is a
match print out the country

     found =1;// set found = 1

   }

 } if(!found){ //if no match is found print statement

   printf("No Match!");

 }

 return 0;

}
```

```
Please Enter an International Dialing Code:
251
The Country that matches the area code you entered is: Ethiopia
```