CSCE 221 Assignment 4 Cover Page

First Name Cody Last Name Williams UIN 924008283

User Name will77868 E-mail address will77868@tamu.edu

Please list all sources in the table below including web pages which you used to solve or implement the current homework. If you fail to cite sources you can get a lower number of points or even zero, read more on Aggie Honor System Office website: `http://aggiehonor.tamu.edu/`

Type of sources				
People	Abhishek Joshi	Bailey Bauman	Samantha Ray	Chris Ridley
Web pages (provide URL)	http://www.cplusplus.com/reference/regex/	http://www.cplusplus.com/reference/regex/ECMAS	https://piazza.com	
Printed material				
Other Sources				

I certify that I have listed all the sources that I used to develop the solutions/codes to the submitted work.
On my honor as an Aggie, I have neither given nor received any unauthorized help on this academic work.

Your Name Cody Williams Date 03-30-17

CSCE 221 — Programming Assignment 4, Spring 2017 (100 points)

Due: March 30th at 11:59 pm

File Parsing and Regex – Assignment Description (100 points)

A very common task in Computer Science is the reading in and parsing of text. One of the most powerful tools at a programmer's disposal to aid in this task is regex (which stands for REGular EXpressions).

1. Read about text manipulation and regex: Programming Chap. 2&3 (PPT)
2. Reference for C++ regex grammar: regex reference
3. (15 points) Compile and run the following code, then answer the questions:
   (a) What is stored in "`matches`"?
      i. A string that matches the specified pattern in regex pattern
   (b) What does "`\d`" mean?
      i. it tells regex to look for a digit

```
#include <iostream>
#include <string>
#include <regex>
using namespace std;
int main() {
   regex pattern{R"(\d\d)"};
   string to_search = "I would like the number 98"
              " to be found and printed, thanks.";
   smatch matches;
   regex_search(to_search, matches, pattern);
   for (auto match : matches) {
      cout << match << endl;
   }
   return 0;
}
```

Note that in C/C++ two subsequent strings as above are combined into one string by the compiler.

   (c) Modify the regex pattern to retrieve a two-digit number and the word `thanks` in the string. Test your pattern for correctness.

      i.

```
#include <iostream>
#include <string>
#include <regex>
using namespace std;
int main() {
    regex pattern{R"(\d\d)"};
    string to_search = "I would like the number 78"
    " to be found and printed, thanks.";
    smatch matches;
    regex_search(to_search, matches, pattern);
    for (auto match : matches) {
        cout << match << endl;
    }
    regex pattern1{R"(\bthanks\b)"};
    smatch matches1;
    regex_search(to_search, matches1, pattern1);
    for (auto match : matches1) {
        cout << match << endl;
    }

    return 0;
}
```

ii.

```
[will77868]@build ~/assignments/Williams-Cody-A4> (16:03:35 03/30/17)
:: g++ -std=c++14 *.cpp

[will77868]@build ~/assignments/Williams-Cody-A4> (16:03:51 03/30/17)
:: ./a.out
78
thanks
```

4. (25 points) Compile and run the following code, then answer the questions:

   (a) What does "\s\S" mean?

      i. It tells regex to look for a white space followed by not a white space

   (b) What is stored in `matches[0]`?

      i. matches[0] stores the entire matched expression

   (c) Why is `matches[1]` different?

      i. matches[1] stores matches within the matched expression (subsequence of the matched expression)

```
#include <iostream>
#include <string>
#include <regex>
using namespace std;
int main() {
   regex pattern{R"(<title>([\s\S]+)</title>)"};
   string to_search = "<html><head>Wow such a header <title>This is a title</title>"
                      "So top</head>Much body</html>";
   smatch matches;
   regex_search(to_search, matches, pattern);
   cout << matches[0] << endl;
   cout << matches[1] << endl;
   return 0;
}
```

   (d) Modify the regex pattern to retrieve only the items inside of the header tag but not inside of the title tag. Test your pattern for correctness.

i.

```
#include <iostream>
#include <string>
#include <regex>
using namespace std;
int main() {
    regex pattern{R"(<head>(.*)<title>.*</title>(.*)</head>)"};
    string to_search = "<html><head>Wow such a header <title>This is a title</title>"
    "So top</head>Much body</html>";
    smatch matches;
    regex_search(to_search, matches, pattern);
    cout << matches[0] << endl;
    cout << matches[1] << endl;
    return 0;
}
```

ii.

```
[will77868]@build ~/assignments/Williams-Cody-A4/part2> (17:29:18 03/30/17)
:: ./a.out
<head>Wow such a header <title>This is a title</title>So top</head>
Wow such a header
```

5. (40 points) Download the following text file: stroustrup.txt

   Write a program using regex that will go through the text file and print out the file name of every hyperlinked powerpoint file. (Hint1: an HTML hyperlink uses the format `<a href="...>...</a>`. Hint2. The powerpoint file extension is `.ppt`)

What to submit to CSNet?

- Your C++ source code with the header block including: your name, user name, section number and e-mail address

- (20 points) A report which should consists of the following parts:
  - The cover page.
  - Assignment number and its description.
    - Assignment 4
    - Description: To practice using regex and writing programs implementing regex.
  - Description of data structures and algorithms used by your program.
    - Data structure: Strings utilized in copmarisons and implementation of regex patterns
    - Algorithm: A simple while loop was utilized to access lines of input files
  - Description of input and output data. List all restrictions and assumptions that you have imposed on your input data and program.
    - Input Data: A text file with html tags containing names of powerpoints
    - Output Data: Names of powerpoints without html tags attached.
    - Restrictions: This program must intake a text file
    - Assumptions: it is assumed that there are html tags within the text file input
  - Write your regex patterns used for parsing the strings in the programs above. Explain their syntax.
    - Problem 3: (\d\d) & (\bthanks\b), Looks for a sequence of 2 digits and the word thanks
    - Problem 4: (<head>(.*)<title>.*</title>(.*)</head>), looks for information inside of head tags but not inside of title tags
    - Problem 5: (<a href=\"(.+)\.ppt), looks for names of powerpoints associated with a html tag
  - What is the purpose of the functions `regex_search()` and `regex_match()`.
    - regex_search(): Checks for a match anywhere in the string
    - regex_max(): Checks for a match at the beginning of the string
  - Which C++ features or standard library classes have you used in your program?
    - iostream
    - string
    - regex
    - fstream
  - Write your conclusion.
    - This assignment was useful in demonstrating the basics of the regex features of c++. From manipulating the given code and building a program to filter powerpoint names from a list of html tags, I have gained a better understaiding of how to utilize regex.