CS 2263 - FR01A

Assignment 3

By Ngoc Phuong Anh Nguyen – 3712361

October 2021

Questions

Question 1: In a few sentences describe the design of your program. Focus on what each of the data structures holds and how each of the functions acts on them.

The program has 3 arrays: a char array named inputArray with size 100000 which stores data from html file, an integer array named countTag with 100 slots that store the number of each tags, and an array of pointers named nametag with 100 slots that stores the pointer to the start of each unique tag.

When the program is begun, it opens a file and copy data into inputArray. When the process is done, the program starts to run through each letter to find out a '<' (this sign is the beginning of tag). Next, it checks the next character if there is a '/' or '!'. If yes, the tags are ignored. Once the beginning of the line which has a valid tag is found, the program use clean() function to clear the line, and leave the tag. For instance, if a tag was "<p style="color:red">", the function would clean it, and transfer it to "<p>", since only "p" is the tag name.

Meanwhile, the valid tags are also count so that the quantities are also stored. When the counting process is done, the program use print() function to print out the tag name and remove the tag signs "<>", and also printout the quantity of each tag.

Question 2: Show the testing of one of the functions using a test program.

temp.c | testing.c | form.html | Sample.html

```c
#include <stdio.h>
#include <stdlib.h>

void print(char *string)
{
  int i = 1;
  while(*(string + i) != '>')
  {
    printf("%c", *(string + i));
    i++;
  }
  printf("\t");
}

int main(int argc, char* argv[])
{
  char a[100];
  char b[100];
  printf("Input Tags: ");
  gets(a);
  printf("Output string: \n");
  print(a);
}
```

Figure 1: Source Code of print() function in a test program.

```
D:\UNIVERSITY\UNB\2021 Fall Term\CS 2263\Assignments\Assignment 3\A3Data\A3Data>.\testing
Input Tags: <html>
Output string:
html
D:\UNIVERSITY\UNB\2021 Fall Term\CS 2263\Assignments\Assignment 3\A3Data\A3Data>
```

Figure 2: Output of the test program.

Question 3: Show the output from running your program on the included HelloWorld.html file.

Question 4: Show the output from running your program on the included Sample.html file.

```c
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>

int clear(char* currentTag)
{
  int i = 0;
  while(*(currentTag + i) !='>' && *(currentTag + i) !=' ' && *(currentTag + i) !='/' &&
  *(currentTag + i) !='\0')
  {
    i++;
  }

  if(*(currentTag + i) != '>')
  {
    *(currentTag + i) = '>';
    int countOpen = 1;
    while(countOpen)
    {
      i++;
      if(*(currentTag + i) == '\0')
      {
        break;
      }
      if(*(currentTag + i) == '<')
      {
        countOpen++;
      }
      else if(*(currentTag + i) == '>')
      {
        countOpen--;
      }
      *(currentTag + i) = ' ';
    }
  }
  return 0;
}
```

Figure 3: Source Code of htags.c

```
37
38  bool compare(char* string1, char* string2)
39  {
40    int i = 0;
41    while(*(string1 + i) != '>' && *(string2 + i) != '>')
42    {
43      if(*(string1 + i) != *(string2 + i))
44      {
45        return false;
46      }
47      i++;
48    }
49    return *(string1 + i) == *(string2 + i);
50  }
51
52  int getIndex(char *currentTag, char **tagNameArray, int tagArraySize)
53  {
54    int i = 0;
55    while(i < tagArraySize)
56    {
57      if(compare(currentTag, *(tagNameArray + i)))
58      {
59        return i;
60      }
61      i++;
62    }
63    return tagArraySize;
64  }
65
66  void print(char *string)
67  {
68    int i = 1;
69    while(*(string + i) != '>')
70    {
71      printf("%c", *(string + i));
72      i++;
73    }
74    printf("\t");
75  }
```

Figure 4: Source Code of htags.c

```c

int main(int argc, char* argv[])
{
  FILE *file;
  char inputArray[100000];
  int pointer = 0;
  int j = 0;

  if(argc == 1)
  {
    file = stdin;
  }
  else
  {
    file = fopen(argv[1],"r");
  }

  if(file == (FILE*)NULL)
  {
    fprintf(stderr, "Unable to open file %s\n", argv[1]);
    return EXIT_FAILURE;
  }

  char c = fgetc(file);

  while(c != EOF)
  {
    *(inputArray + pointer) = c;
    pointer++;
    c = fgetc(file);
  }
  *(inputArray + pointer) = '\0';

  int countTag[100];
  char *nameTag[100];
  int tagArraySize = 0;
  for(pointer = 0; *(inputArray + pointer) != '\0'; pointer++)
  {
    if(*(inputArray + pointer) == '<' && *(inputArray + pointer + 1) == '!')
```

Figure 5: Source Code of htags.c

```
    {
      clear(inputArray + pointer);
    }
    else if(*(inputArray + pointer) == '<' && *(inputArray + pointer + 1) != '/' && *(inputArray
    + pointer + 1) != '\0')
    {
      clear(inputArray + pointer);
      int index = getIndex(inputArray + pointer, nameTag, tagArraySize);
      if(index == tagArraySize)
      {
        tagArraySize++;
        *(nameTag + index) = inputArray + pointer;
        *(countTag + index) = 1;
      }
      else
      {
        *(countTag + index) += 1;
      }
    }
  }

  pointer = 0;
  while(pointer < tagArraySize)
  {
    print(*(nameTag + pointer));
    printf("%d\n", *(countTag + pointer));
    pointer++;
  }

  fclose(file);
  return EXIT_SUCCESS;
}
```

Figure 6: Source Code of htags.c

```
D:\UNIVERSITY\UNB\2021 Fall Term\CS 2263\Assignments\Assignment 3\A3Data\A3Data>gcc -c htags.c

D:\UNIVERSITY\UNB\2021 Fall Term\CS 2263\Assignments\Assignment 3\A3Data\A3Data>gcc -c htags.c

D:\UNIVERSITY\UNB\2021 Fall Term\CS 2263\Assignments\Assignment 3\A3Data\A3Data>gcc -o htags htags.c

D:\UNIVERSITY\UNB\2021 Fall Term\CS 2263\Assignments\Assignment 3\A3Data\A3Data>.\htags < HelloWorld.html
html    1
head    1
meta    1
title   1
body    1
p       1

D:\UNIVERSITY\UNB\2021 Fall Term\CS 2263\Assignments\Assignment 3\A3Data\A3Data>.\htags < Sample.html
html    1
head    1
meta    1
title   1
body    1
strong  1
ol      1
li      2
blink   1
p       2
```

Figure 7: Output of Question 3 and 4

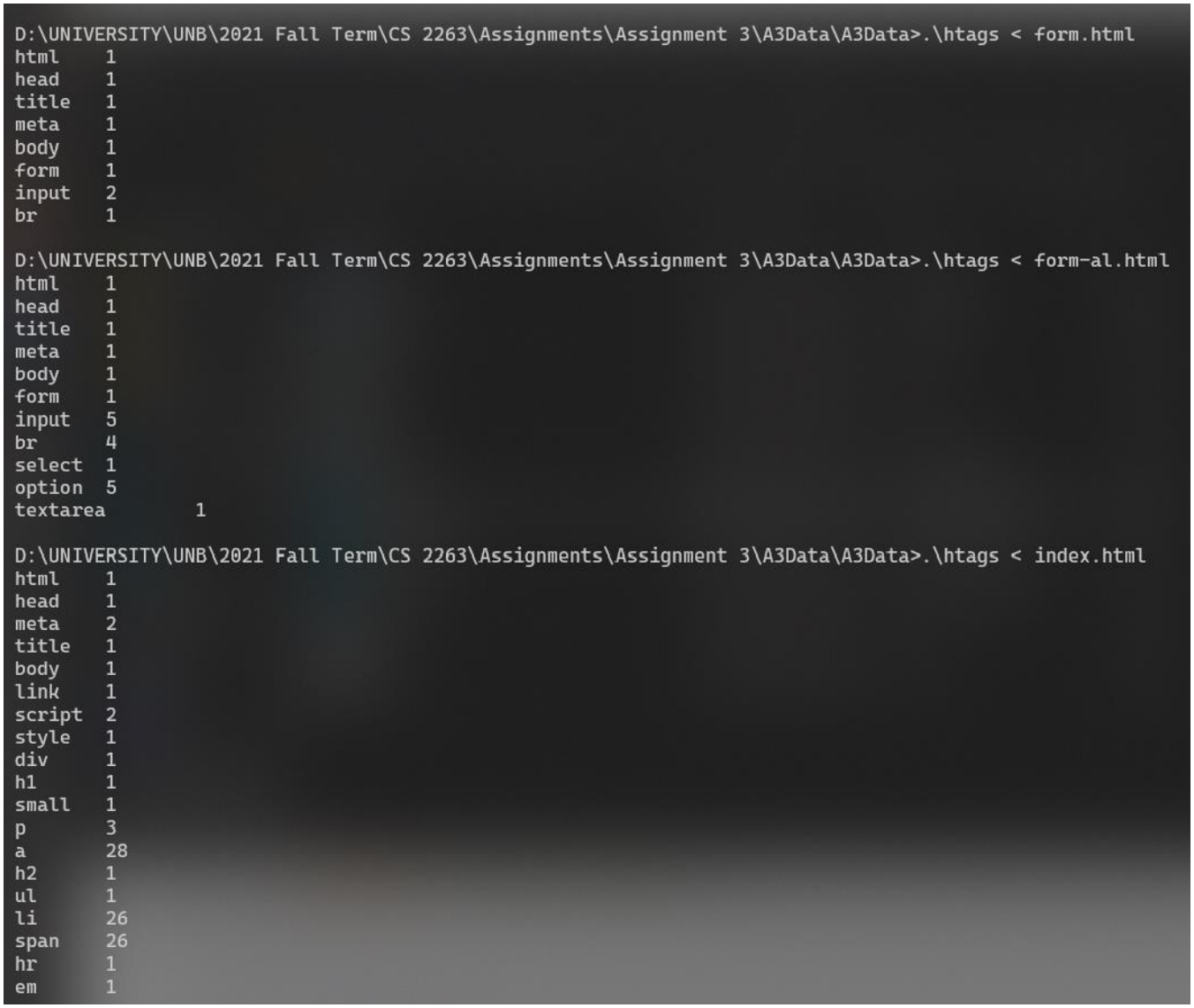

Figure 8: Output of htags.c using form.html, form-al.html and index.html