Practice Exam I (110pts)

Debugging: Find and correct each bug:(10pts) Rewrite correct code here (you may pretend you are already inside main): (2pts each)

1.

```
#include <iostream>
using namespace std;

int main(){
  return "Hello world!\n";
}
```

2.

```
#include <iostream>

int main(){
  cout << “Hello World!\n”;
  return 0;
}
```

3.

```
#include <iostream>
using namespace std;

int main(){
  char c = "Hello World!\n";
  cout << c;

  return 0;
}
```

4.

```
#include <iostream>
using namespace std;

int main(){
  for(i = 0; i < 10; i++){
    cout << i << endl;
  }
  return 0;
}
```

5.

```
#include <iostream>
using namespace std;

int main(){
  int x = 3;
  if(x < 4 && < 5){
     cout << “x is tiny”;
  }
  return 0;
}
```

Use Existing Code (10pts): Write a main function that creates a random username by selecting a random adjective and random noun from two files and concatenating the words together separated by an underscore. So for instance, if the string “hilarious” is randomly selected from the first file and the string “hamster” is randomly selected from the second file, the username that is sent to standard output should be “hilarious_hamster”. In each file, there will be one word per line, but the number of lines in the file will be unknown and vary per use of the program.

Call the functions defined below in your main() function to accomplish the goal. The number of lines in each input file can vary, so make sure you programmatically count the number of lines in each file before randomly choosing a line to pull a word from so that you don’t accidentally try to pick a word passed the number of lines in the file. Also, make sure you pull adjectives and nouns from their respective files as documented in the code below.

```
int getLineCount(string filename)
{
  ifstream in;
  in.open(filename);

  string line;
  int line_count = 0;

  while(getline(in, line)){
    line_count++;
  }

  in.close();

  return line_count;
}
```

```
string getAdjective(int max){
  return getWord(max, ADJ_FILE);
}


string getNoun(int max){
  return getWord(max, NOUN_FILE);
}
```

```
string getWord(int max, string filename){
  string word;
  unsigned int s = time(0);
  int min = 1;
  ifstream in;

  in.open(filename);
  srand(s);

  int r = (rand() % (max - min + 1)) + min;

  for(int i = 0; i < r; i++){
     in >> word;
  }

  in.close();

  return word;
}
```

```
#include <iostream>
#include <fstream>
#include <cstdlib>
#include <ctime>

using namespace std;

int getLineCount(string);
string getWord(int, string);
string getAdjective(int);
string getNoun(int);

const string ADJ_FILE = "adjectives.txt";
const string NOUN_FILE = "nouns.txt";

int main(){
  //write your answer here inside main by calling the functions above










}
```

Answer the questions below: (20pts – 2pts each)

1. What is the signature of a function, discuss in terms of the function head void printScreen(int x, int y, char c).

2. Given the operation 11011100 | 00111010 what is the output?

3. What is the difference between pass by value and pass by reference?

4. What is an algorithm?

5. What is the difference between RAM and secondary storage (hard drive)?

6. What is the difference between a compiled and interpreted language?

7. What is the difference between a binary and a text file?

8. What is the return type of main?

9. Under what conditions will p && q return true?

10. When should you use a while loop instead of a for loop?

Trace the programs: (30pts – 15pts each) Please write the output of each program in the boxes to the left, remember to keep track of your variable states using registers to aid your concentration. Assume using namespace std and any required libraries are included in the program

1.
#include <iostream>
using namespace std;

```
int f1(int x, int y){
  x = y + 11;
  cout << “inside... “ << endl;
  cout << x << “ “ << y << endl;
  cout << “returning...” << endl;
  return x;
}


int f2(int x, int y){
  return f1(y, x);
}


int main(){
  int x = 1;
  int y = 8;
  int a = 3;
  int b = 4;

  cout << “call 1: “ << endl
       << f1(x, y) << endl;
  cout << x << “ “ << y << endl
       << endl;

  cout << “call 2: “ << endl
       << f1(f1(a, b), f1(b,a))
       << endl << endl;

  cout << “call 3: “ << endl
       << f2(x, y)
       << endl;

  return 0;
}
```

2.

```
#include <iostream>
using namespace std;


bool r1(char &g, char &h){

  static int q = 0;
  q++;

  cout << “q: “ << q << endl;

  if(g != ‘ ‘ || h != ‘ ‘){
    cout << “inside if” << endl;
  } else if (g != h){
    cout << “inside else if” << endl;
  } else {
    cout << “inside else” << endl;
  }

  if(g == ‘A’){
    cout << “inside 2nd if” << endl;

    if(h == ‘B’){
       cout << “inside nest if” << endl;
    } else {
       cout << “inside nest else” << endl;
    }
  }

  char s = g;
  g = h;
  h = s;

  return g == h;
}

int main(){
  char s = ‘#’;

  for(int i = 0; i < 3; i++){
    for(int j = 0; j < 2; j++){
      cout << s;
    }
    cout << endl;
  }

  char g = ‘ ’;
  char h = ‘p’;

  cout << r1(g, h) << endl;

  cout << “g: “ << g << endl
       << “h: “ << h << endl
       << “s: “ << s << endl;
  char n = ‘A’;
  char m = ‘C’;
  cout << r1(n, m) << endl;

  return 0;
}
```

Write the program requested below: (40pts)

1. Write a program that reads the contents of data.txt and converts it to a sentence that is sent to the file results.txt. Each number in data.txt should be converted to its corresponding ASCII character. Each line in data.txt should be converted to a word. In the output file, results.txt, each word converted from data.txt should be separated by a space. So for example, the input from data.txt should be converted to the sentence “Eat my Shorts!” with 69,97, and 116 corresponding to the word “Eat” and so forth. Your program should work for any file with lines of numbers separated by spaces. Your program cannot assume knowledge of the contents of data.txt ahead of time, including the number of lines in the file or how many numbers are on each line.

Please break your program up into functions. You should have one function that converts a string of digits into an ASCII character. The function should be named convertInt(). convertInt() should take a string as input and return a char. So for instance if the string “65” is provided as input, the output should be the character “A”.

You should have a second function processLine() that takes a string as input and returns a string as output. The input string should be a single line from data.txt. The output should be the word that is encoded by that line. So for instance, if the function takes “69 97 116” as input, it should output the word “Eat”. The processLine() function should call the convertInt() function inside of itself in order to perform its ASCII character conversions.

Example data.txt:

```
69 97 116
77 121
83 104 111 114 116 115 33
```

Example results.txt:

```
Eat My Shorts!
```

Extra Space: