**CS421 - Krell - HW1B (based on Week2b) Recognizer vs. Generator – Programs**

**Answer next to each ?\*\*?**

**=====================================================================**

**DUE: Week 3 Friday 2/12 before midnight (11:55pm)**

**TOTAL: 30 pts**

**\*\*NAME: Jennifer Luu**

**======================================================================**

**To receive points for any programming HW for CS421:**

* **Read @instructions file first.**
* **Your programs must be compilable and tested using g++ on empress.**
* **Your outputs must be a screen dump, or a "script" (.txt file)**
* **And it must have the line for compiling the files!! (i.e. g++)**
* **The state of the program statement must always be provided!!!**
* **Even if your program does not compile, submit the test results.**
* **Even if your program does not run well, submit outputs for parts that are working.**

**Files are on empress.csusm.edu in folder /cs/cs421LK/CS421Progs/HW1B\_RecGen**

**A) Write a recognizer in C++ for L = {x | x is a binary number} [10 pts]**

**My recognizer.cpp must be used.**

**main: Given a string (from** ε**) cined from the user, pass it to**

**the recognizer function.**

**Cout “YES IN L” or “NO NOT IN L” depending on what was returned.**

**recognizer function: Should return TRUE or FALSE checking each character**

**to make sure it is 0 or 1. (leading 0’s are allowed).**

**Testing: test with 0, 1, 00110, 2, 02, 31 in this order. 🡺Routput.txt**

**Answer the following regarding your program**

**\*\* State of the program: <required to be graded. Explain completeness, correctness, bugs, etc.>**

**The program was able to run smoothly without any errors. My output file (recOutput.txt) matches the Routput.txt file.**

**Graded on: [total 10] I will fill out this table.**

**Correctness:**

**Comments:**

**Test results:**

**Following the instructions:**

**B) Write a generator in C++ for L = {x | x is a binary number}.[20 pts]**

**My generator.cpp must be used.**

**main: It should create each string over E = {0,1,2} systematically**

**(short to long) and pass each string to the recognizer function**

**created in Part a) above.**

**Only those strings for which the recognizer returned TRUE**

**should be displayed.**

**i.e.**

**create 0 --> returns TRUE ---> display 0**

**create 1 --> returns TRUE ---> display 1**

**create 2 --> returns FALSE**

**create 00 --> returns TRUE ---> display 00**

**create 01 --> returns TRUE ---> display 01**

**create 02 --> returns FALSE**

**create 10 --> returns TRUE ---> display 10 and keep on going**

**[The challenge here is to figure out how to create**

**all strings using 0, 1 and 2 systematically, with no repeated**

**strings. Hint CS311 HW1 using a queue.]**

**recognizer function: the same one from part A) (copy and use)**

**Testing: The user must somehow interactively terminate**

**the execution of the program after at least 20 binary strings**

**have been displayed although your program should be able to keep on going until the queue overflows. 🡺Goutput.txt**

**Answer the following regarding your program**

**\*\* State of the program: <required to be graded. Explain the state of your program here, bugs etc.>**

**The program was able to run smoothly without any errors. I implemented the option for the user to exit the program using “exit” in addition to using Ctrl+C. My output file (genOutput.txt) matches the Goutput.txt file.**

**Graded on: [total 20] I will fill out this table.**

**Correctness:**

**Comments:**

**Test results:**

**Following the instructions:**

**Submit 5 files to assignment HW1B on Cougar Courses:**

1. **Submit both the well-commented program recognizer.cpp**
2. **Submit both the well-commented program generator.cpp.**
3. **Submit the output from recognizer created on Empress (Routout.txt file).**
4. **Submit the output from generator created on Empress (Goutput.txt file).**
5. **Submit this sheet as well.**

**Always be extra careful and double check what you are submitting.**