

1 Project: Maze Generator.

Implement the union-find based maze generating algorithm discussed in Section 8.7 of the text. Your algorithm should create mazes on grids of size n by m (for arbitrary n and m). You should also implement a function to print the maze. You can print them using “ASCII art”. For example, Figure 8.23, could be printed out as:

```
#####  
      # #  
### ##### #  
# #      #  
# # ##### #  
# # # # #  
# ### # # #  
# #      # #  
### # #####  
# #  
#####
```

2 Tasks.

1. Implement the Union-Find ADT with path-compression and union-by-rank or union-by-size.
2. Implement the maze generating algorithm from the text.
3. Implement a print function (e.g., to “ASCII art”).
4. Combine these into a program that prints a random maze.

3 Logistics.

This programming assignment is optional. If you choose to do this assignment, your total grade for programming assignments will be taken out of 60 total points (20 for each assignment), and each assignment will be weighted equally. If you choose not to do this assignment, your total grade for programming assignments will be taken out of 40 total points (20 for each assignment), and each assignment will be weighted equally.

This programming assignment will be graded out of a total of 20 points. It is due on 12/10/08 (at midnight).

Submitting your code. The T-Lab (Tech F252) is available for your programming needs. Your programs should compile under `g++`. To submit your program send email the source code by email to the TA. Use the subject line “SUBMIT PROG3”. Do not submit executables. If you use a

makefile, submit the makefile. If you compile your program directly on the command line, specify the command line in your email.

Grading guidelines. You will be graded on your ability to write efficient code. You will be graded on your ability to write reasonable C++ code. You will be graded on your ability to implement the required tasks. You will be graded on your ability to manage memory (i.e., be careful of *memory leaks* and other bugs with memory usage).

Resources. You may consult your text book or other books on C++ and data structures. You may not use the Standard Template Library. You must not copy code from anywhere. You may talk with your classmates about the project at a high level, but your implementations must be 100% original. You may consult with the instructor and TA on any aspect of the project.