Jumble Solver

By: Michael Lehmann & John Fullard

CSE 30331 – Data Structures
2008
Previously

• The project started differently...
Issues

- Problem with the previous method...
  - slow
Changes

• Indexing of letters.
• Count occurrences.
• Store as numbers.
void OccurCnt(const char* ins, int counter[26])
{
    assert(ins); //assert is just good programming practice
    int cnt;
    int len = strlen(ins);
    //this line sped loading the dictionary up by over 20 minutes
    memset(counter, 0, 26 * sizeof(int));
    for (cnt = 0; cnt < len; cnt++)
    {
        if (ins[cnt] >= 'a' && ins[cnt] <= 'z')
        {
            counter[ins[cnt] - 'a']++;
        }
    }
}
Opening Image:

Jumble Solver
By: Michael Lehmann & John Fullard
Doing it's job...

Jumble Solver
By: Michael Lehmann & John Fullard

Word to Solve:
llulfll

Solve!
Clear

| l | 1 |
| l | 2 |
| u | 1 |
| f | 1 |
| l | 3 |
| i | 1 |
| f | 2 |

Solutions:
fulfill
Includes

• QT 4 is needed to run / compile the program.
Questions?