Programming Assignment 1

Assigned: Sept. 1
Due: Sept. 22

Assignment

Write two programs that solves the Best Vertex Cover problem, as described in [Problem Set 1](). The first will use iterative deepening; the second will use simple hill climbing with random restart.

Input

Input should be a text file (or read from standard input) with the following contents.

- Line 1: The budget (a positive integer) and flags: "V" for verbose output or "C" for compact output. For the hill-climbing program, the number of random restarts to run.
- Each vertex. Name (you may assume that this is a single alphabetic character) and cost (a positive integer), 1 per line.
- Blank line
- Each edge: the names of the two ends. 1 per line.

Examples are given below.

Search

The programs should search through the state space until either:
a) A solution is found.
b) In iterative deepening, search to depth K finds no states at depth K. (That is, none of the states at depth K-1 have any successors). Thus, if there is no goal state to be found, the search to depth K will be identical to the search to depth K-1; then the program will terminate.
c) In hill climbing, each hill climb is carried out from a random starting state until it reaches a solution or a local maximum. A random starting point is constructed by going through the list of vertices and accepting or rejecting each with probability 1/2.

In doing the hill climbing part of the assignment, it's a good idea to write your code so there is a module where you can specify the starting state. That way, you can both replicate your code's behavior, which is useful in debugging, and compare it against my samples. However, the code that you submit should construct random starting states.

Output

The program should print out to a text file or standard output:

- If a solution is found, the solution (just a sequence of vertices).
  If no solution is found, then "No solution found"
- If the "verbose" flag is set then a trace of the search sequence, as illustrated below.

Coding

Programs will be accepted in Java or Python. Check with me about other languages.

Submit

The source code for the two programs and a README file how to compile and run your code.

Grading

Grading: 8 points for correctly running code. 2 points for well-written code. A programming assignment count equally with a problem set in computing the overall grade.

Late policy

Programming assignments due at the beginning of class on the due date. I will accept assignments late with a penalty of 1 point out of 10 for each week late (fractions of a week are rounded up) up to 2 weeks late.

There are some additional sample inputs and outputs linked to the assignment on the web.

Sample Inputs and Outputs

There are additional sample I/O pairs linked, for iterative deepening and for hill climbing.

Input for iterative deepening program, for graph 1. Budget=25

```
25 V
A 3
B 5
C 6
D 10
E 13

A B
A C
B D
C E
D E
```

Output

```
Depth = 1
A Cost=3.
B Cost=5.
C Cost=6.
D Cost=10.
E Cost=13.

Depth=2
A Cost=3.
A B Cost=8.
A C Cost=9.
A D Cost=13.
A E Cost=16.
B Cost=5.
B C Cost=11.
B D Cost=15.
B E Cost=18.
C Cost=6.
C D Cost=16.
C E Cost=19.
D Cost=10.
D E Cost=23.
```

```
E Cost=13.

Depth=3
A Cost=3.
A B Cost=8.
A B C Cost=14.
A B D Cost=18.
A B E Cost=21.

Found solution A B E Cost=21.
```

The input for hill climbing is the same, except that you have to indicate the number of random restarts on the top line. E.g.

```
15 V 2
```

for a budget of 15, with two random restarts, verbose output.

In verbose output for hill climbing, you should indicate the value of the Error function as well as the cost for a state, and show all the neighbors that are being tested. (The order in which you enumerate neighbors doesn't matter.)

Sample output

```
Randomly chosen start state: E
E Cost=13. Error=11
Neighbors:
A E Cost=16. Error=6
B E Cost=18. Error=6
C E Cost=19. Error=12
D E Cost=23. Error=14
{} Cost=0. Error=27

Move to A E. Cost=16. Error=6
Neighbors
E Cost=13. Error=11
A B E Cost=21. Error=6
A C E Cost=22. Error=12
A D E Cost=26. Error=11
A Cost=3. Error=21

Search failed

Randomly chosen start state A B C
A B C Cost=14. Error=10
Neighbors
B C Cost=11. Error=10
A C Cost=9. Error=15
A B Cost=8. Error=16
A B C D Cost=24. Error=9
A B C E Cost=27. Error=12

Move to A B C D Cost=24. Error=9
Neighbors
B C D Cost=21. Error=6
A C D Cost=19. Error=4
A B C Cost=14. Error=10
A B D Cost=18. Error=9
A B C D E Cost=37. Error=22

Move to A C D Cost=19. Error=4
Neighbors
C D Cost=16. Error=4
```

```
A B C D Cost=24. Error=9
A D Cost=13. Error=6
A C Cost=9. Error=15
A C D E Cost=32. Error=17

Search failed

No solution found.
```