MATH 567 Project 2

Xiyu Xie 11190877

0. Program are Indexing are from 0, (instead of 1)

1. Methodology

Algorithms are implemented as taught in class. With following points worth of mentioning:

About Modified Greedy:

(1) Score2 is ScoreG with K=1

(2) Score1 is ScoreG with K=1, Indicator for t- hard to cover counts

About Preprocessing

(0) A preprocessing call runs all three condition in sequence.

(1) Condition II: Remove pole which cover all meters the anther pole covers

Let M(P) is the meter set covered by pole P

This conditions is saying:

if M(p1) <= M(p2)

remove p1

(2) Condition III: Remove meter which all the poles cover it also cover anther meter

Let P(M) be the pole set covering meter M

This condition is saying

if P(m1) <= P(m2)

remove m2

(3) Subset Testing is implemented by Bitmap

Let array of integer represent poles covering a meter (vice versa)

Bit position j is 1 if pole j covers this meter, (otherwise 0)

Let B(M) denote the bitmap for meter M, for example.

Notice, P(m1) <= P(m2)

if and only if

B(m1) & B(m2) == B(m1)

where & is bitwise AND operation.

(4) For this reason, this project is implemented in C

2. Implementation Details

       type.h contains all data and explanation used in this program

       include.h contains function unit and explanation of this program

       greedy.c or ModifiedGreedy.c explained basic structure of this program

       readme.txt gives additional information about how to run this program

3. Possible Improvement

       Randomized Preprocessing in time and choice of function may have better performance

       Code for greedy could be combined with modified greedy, but not for efficiency

4. Results

Folder "output" contains output files for each run summarized below

Each file are named as "data.method", for example,

       ph1.greedy is greedy algorithms for phase1

       ph1.pgreey is greedy algorithm with preprocessing for phase1

Each file ouput:

       removed pole and meter by preprocessing

       cleanuped poles

       Selected poles

       summary information in following table

G_Score Methods has best result, even with K=2 only

			Value	Feasible	Time/Sec	Cleaned Any
No preprocess	Greedy	Phase1	24	Yes	0	No
		Cap360	617	Yest	0.36	Yes
	Score1	Phase1	25	Yes	0	No
		Cap360	632	Yes	1.21	Yes
	Score2	Phase1	25	Yes	0	No
		Cap360	618	Yes	0.99	Yes
	ScoreG K=2	Phase1	23	Yes	0	No
		Cap360	615	Yes	2.63	Yes

			Value	Feasible	Time/Sec	Cleaned Any
Preprocess per each selection	Phase1	Greedy	23	Yes	0	No
		Score1	25	Yes	0.01	No
		Score2	25	Yes	0.01	No
		ScoreG K=2	23	Yes	0.01	No
Preprocessed Only Once in the middle of Run	Cap360	Greedy	614	Yes	40.59	Yes
		Score1	624	Yes	52.32	Yes
		Score2	611	Yes	52.52	Yes
		ScoreG K=2	601	Yes	53.76	Yes
		ScoreG K-3	587	Yes	57	Yes