COMPUTING LAB-1

ASSIGNMENT 3

GAUTAM SAGAR, 21CS60R15

PART 1: CODES WITHOUT REPETITION OF COLOURS:

For 4 positions and 8 colours a total of 32 variables are used to represent positions of colours.

Position encoding for colours: C={R,B,G,Y,O,P,W,K}

Red(R) : 1, Blue(B): 2, Green(G): 3, Yellow(Y): 4, Orange(O): 5, Purple(P): 6, White(W):7, Black(K): 8

32 variables are divided among these 8 colors.

Variable number = (positional encoding for colour-1)*4+position

Eg: For a Code BYWO the encoding is 5 14 27 20

Now, Constraints are defined using these encodings with the repetition of colours not allowed.

Eg. If Red is in position one then no other colour can occupy first position in the code. This is encoded as(In DIMACS Format):

-1 -2 0 //No repetition
-1 -3 0 //No repetition
-1 -4 0 //No repetition
-1 -5 0 //Only Red at the first position
-1 -9 0 //Only Red at the first position
-1 -13 0 //Only Red at the first position
-1 -17 0 //Only Red at the first position
-1 -21 0 //Only Red at the first position
-1 -25 0 //Only Red at the first position
-1 -29 0 //Only Red at the first position

Because the contrapositive statements are also true thus constraints like -29 -1 0 and -2 -1 0 are not added.

Furthur 4 more constraint i.e. 33,34,35,36 were added to ensure that the colour at kth position satisfy positional encoding.

PART 2: CODES WITH REPETITION OF COLOURS

The only difference between part 1 and part 2 is that repetition of colours is allowed in part 2. This allows us to remove some constraints from DIMAC file.

Eg For Red color, constraints to be removed are:

-1 -2 0 //No repetition
-1 -3 0 //No repetition
-1 -4 0 //No repetition

These constraints means that if there is Red at first position then no Red is allowed on any other position. So removing these constraints allows us to repeat colours.

BOOLEAN CONSTRAINTS ADDED BY THE CODEBREAKER:

Code maker gives feedback or hint for every guess the codebreaker makes in terms of number of black and white pegs. Thus for each hint the codebreaker receives, it adds some constraints. For different values of black these constraints are given below:

Let A be the colour at the first position, B be the colour at the second position, C be the colour at the third position and D be the colour at the fourth position.

B-i represents that the number of black pegs=i.

Not: ~

OR: ||

AND: &

B-0: All guesses are false => ~A&~B&~C&~D

B-1: Constraints are given below in the CNF Form:

1. ~A || ~B

2. ~A || ~C

3. ~A || ~D

4. ~B || ~C

5 ~B || ~D

6. ~C || ~D

B-2:

1. ~A ||~B||~C
2. ~A||~B||~D
3. ~A||~C||~D
4. A||B||C
5. A||B||D
6. A||C||D
7. ~B||~C||~D
8. B||C||D

B-3:

1. ~A ||~B||~C||~D
2. A||B
3. A||C
4. A||D
5. B||C
6. B||D
7. C||D

B-4: All are true => A&B&C&D