HW 11

2. Грамматика $\ll S- > aSbbbb|aaaSbb|c \gg$ задаёт язык
$L = \{a^{n+3m}\ c\ b^{4n+2m}\}$, где n и m неотрицательные целые числа - количество применений первого и второго правил соответственно
Очевидно, что порядок правил неважен, мы можем вначале применить все первые правила, затем все вторые и получим то же самое слово
Поэтому можно переписать нашу грамматику в такой вид:
$S- > aSbbbb|T$
$T- > aaaSbb|c$
А это уже однозначная грамматика, так как по количеству букв a и b мы можем восстановить количество применений первого и второго правил:
Пусть $w = a^n\ c\ b^m$ и мы применили x раз первое правило и y раз второе
Тогда $x + 3y = n$ и $4x + 2y = m$
Эта система имеет единственное решение:
$x = (3m - 2n)/10$ и $y = (4n - m)/10$
Значит дерево вывода задаётся однозначно

3. $F- > \epsilon|aFaFbF$
Заметим, что каждый раз, когда мы используем какое-нибудь правило и на свет рождается новая буква b, то обязательно где-то слева от неё рождаются две буквы a
Поэтому на любом префиксе букв a хоть бы в 2 раза больше чем букв b
Ну и так как мы всегда добавляем ровно 1 b и 2 a, то суммарно в строке букв a ровно в 2 раза больше

4. Давайте сократим грамматику $F- > a|bF|cFF$ так, чтобы она принимала только слова из второй грамматики
Заметим, что в грамматике :
$K- > aM|cM$
$M- > aK|bK|\epsilon$
Нетерминалы чередуются
K допускает только буквы a и c, а M только буквы a и b
И так как начинаем мы с K, а заканчиваем M, то все слова состоят из нечетного количества букв
То есть это слова вида: $(a|c)(a|b)(a|c)(a|b)...(a|c)$
То есть в нашей первой грамматике нам нужно чередовать правила bF и cFF так, чтобы не оказалось bb и cc и в слове было нечётное количество букв
Получается такая грамматика:

Нетерминалы C_0, C_1 будут отвечать за переходы, которые начинаются с буквы c и содержат суммарно чётное и нечётное количество букв соответственно
Аналогично B_0 и B_1

Грамматика:

$C_1 \to a|cB_0B_0|cB_1C_1$
(других переходов нету, потому что во всех остальных будет bb либо cc
Например не подходит переход $C_1 \to cC_1B_1$, так как первая буква c будет стоять рядом со второй буквой c из C_1)
$C_0 \to cB_0B_1|cB_1C_0$
$B_1 \to a|bC_0$
$B_0 \to bC_1$

P.S. с написанным алгоритмом CYK так легко проверять свой ответ, кааайф :)