

АМВ-3

Волынцев Дмитрий 676 гр.

28 февраля 2017

Задача 1

1) Все вершины двудольного графа разделены на две группы, в каждой из которых нет смежных вершин. Тогда в нем нет трех смежных вершин, а значит и треугольников. Таким образом данный язык пустой, следовательно можем построить машину Тьюринга/автомат, разрешающий его за полиномиальное время, ч.т.д.

2) Будем проверять наличие пути между вершинами с помощью алгоритма поиска в глубину (использовался в дискретном анализе) и в соответствии с этим принимать/отвергать граф. Поиск в глубину работает за полиномиальное время, значит и МТ будет работать за полином, значит язык принадлежит классу P , ч.т.д.

3) Будем двигаться по матрице смежности слева направо и сверху вниз. Найдя единицу (матрицу порядка 1 из единиц), проверим три соседние с ней по ходу движения клетки. Если там все единицы, записываем в искомую клетку 2 вместо 1. Продолжив проверять остальные клетки, сделаем в худшем случае $3 * (n - 1)^2$ шагов (последние столбец и строка не проверяются). На следующем проходе алгоритма ищем двойки и проверяем клетки на расстоянии 2, то есть $5 * (n - 2)^2$ шагов. Повторяем $n - 2018$. Таким образом всего $3 * (n - 1)^2 + 5 * (n - 2)^2 + \dots < (\frac{2 * 3 + (n - 2019) * 2}{2} * n) * n^2 < n^4$, то есть время работы $O(n^4)$ - полином, ч.т.д.

Задача 2

Задача 3

Задача 4

Пусть $L \in P$ и M - МТ, разрешающая L за полином. Пусть M_* - такая МТ, что: на входе x
 $pos = \{0\}$ - позиции, на которых заканчиваются слова из L

```
for i = 1 to |x|
  for j ∈ pos
  {
    if M(x[j + 1, ..., i]) == accept
    {
      if i == |x|
        return accept
      pos ∪ = {i} - добавляем в pos i
    }
  }
return reject
```

Если удастся разбить входное слово на подслова из L , то M_* принимает его, иначе отвергает, а значит разрешает L^* . При этом работает за полиномиальное время, а значит $L^* \in P$ (идея взята в Викиконспектов)

Задача 1-доп

$$T(n) = T(\alpha n) + T((1-\alpha)n) + \Theta(n) = T(\alpha^2 n) + 2T(\alpha(1-\alpha)n) + T((1-\alpha)^2 n) + \Theta(n) = \\ \dots = \sum_{i=0}^k C_k^i T(\alpha^{k-i}(1-\alpha)^i n) + \Theta(n)$$

Пусть $\beta = \max(\alpha, 1-\alpha)$, оценим дерево:

$$\Theta(n) \log_{1/(1-\beta)} n \leq T(n) \leq \Theta(n) \log_{1/\beta} n$$

Таким образом $T(n) = \Theta(n * \log n)$