

1 Обязательные задачи

1. За $O(n + m)$ можно проверить, является ли a подпоследовательностью $b \rightarrow$ эта задача $\in P$.
2. Делаем бинарный поиск по ответу. Затем пусть у нас есть n клозов, и мы хотим проверить, можно ли удовлетворить k из них. Делаем схему сложения, где входы – значения клозов, а выходов $\log n$. Тогда k однозначно определяет, какие из этих $\log n$ должны быть равны 1, то есть эти гейты надо выполнить. Запускаем CIRCUIT-SAT.
3. Напишем на i -ом предмете число m^i , где m – (число множеств $B_j + 1$). Весом множества будет сумма весов его элементов. Тогда если множества покрывают U без пересечений, то их вес будет $= \sum_{i=1}^n m^i$, а иначе не будет, потому что чтобы получить i -ую степень числа m , надо иметь m экземпляров $(i-1)$ -й степени, а у нас их максимум $m-1$, т.к. множеств $m-1$.
4. Решим VERTEX COVER через SET-COVER. Поставим каждой вершине в соответствие множество рёбер, которые она покрывает, и решим SET-COVER. Успех.
Update: Подсказка для decision-версии – покрывающий набор множеств.
5. Сведём SET-COVER к HITTING-SET. Поставим в соответствие каждому элементу множество множеств, в которых он лежит, а наши множества сделаем элементами. Тогда если раньше элемент был в множестве, то теперь наоборот. Решим HITTING-SET.
Update: Подсказка для decision-версии – покрывающий набор элементов.
6. Сведём HAMCYCLE к нашей задаче. Существующим рёбрам поставим вес 0, несуществующим – 1. Если у минимального по весу гамильтонова цикла вес 0, то он существует, иначе нет.
Decision – существует ли путь веса $\leq W$. Понятно, что она NP-hard (то же сведение HAMCYCLE к ней), осталось показать, что она в NP. Подсказка – сам путь \rightarrow она в NP.
- 7.
8. 3-SAT \in NP, но если ETH, то 3-SAT $\notin P$. чтд.

2 Дополнительные задачи

1.
 - 1) PARTITION \rightarrow JOB-SCHEDULING. У нас будет два рабочих, а заказ – это предмет, и его время выполнения – вес предмета. Минимизируем время

выполнения – минимизируем разность между одним множеством и другим, если эта разность 0 – успех.

2) SUBSET-SUM \rightarrow PARTITION. Пусть вся сумма весов – S, а мы хотим найти подмножество веса W. Добавим предмет веса $K = S - 2W$. Поскольку найти подмножество веса – это то же самое, что найти подмножество веса $S - W$, мы можем выбрать $W \leq S/2$. Теперь сделаем PARTITION. Выкинем предмет веса K оттуда, куда он попал. Тогда оставшееся множество будет веса W. Ура.

2.

3. Известный критерий, что p простое – $\exists a : a^{p-1} = 1 \pmod{p}$, но $\forall q : q | (p-1)$ и q – простое, $a^{\frac{p-1}{q}} \neq 1 \pmod{p}$. Тогда подсказка – это такое a и разложение числа p - 1 на простые множители. Дальше легко проверить всё, что нужно, бинарным возведением в степень. Осталось проверить корректность подсказки, то есть проверить, правда ли данные нам числа простые. Давайте рекурсивно запустим PRIME от них. Тогда пусть $k = \log p$. $T(k) = Poly(k) + T(x_1) + T(x_2) + \dots + T(x_m)$, причём $\sum x_i \leq k - 1$, т.к. p - 1 чётное и, следовательно произведение всех нечётных множителей p - 1 хотя бы в два раза меньше p. Давайте проверим, что такая рекурсивная формула задаёт нам полином от k.