Билеты

№1

Алфавит = $\underset{\text{конечная}}{\Sigma}$ символов

слово = конечная пос-ть символов из алфавита.

Словарь = Σ^* произвольных символов из алфавита

№2

Конструктивный объект – объект, информация о котором может быть записана с помощью конечного кол-ва символов.

Конструктивное пространство – множество кон-ых объектов одного типа.

$$\boxed{\text{КП } 1} \xrightarrow{F} \boxed{\text{КП } 2}$$

Функция F – вычислимая, если $F: \text{КП}1 \to \text{КП}2$, причем нужно конечное кол-во шагов для описания

Пример:

КП1 – множество натуральных чисел

КП2 – множество целых

$x \in КП1$ $F(x) \in КП2$

$$F(x) = (2 \cdot x - 1) \cdot (-1)$$

Алгоритм:

$$x \xrightarrow{\cdot 2} 2x \xrightarrow{-1} 2x - 1 \xrightarrow{\cdot(-1)} F(x)$$

№3

ДКА:

Q – множество кон-ых состояний

$F \subset Q$ – множество конечных состояний для прекращения работы

q_0 – начальное состояние.

На вход – слова из Σ

q_k – текущее состояние

$\bar{q}_i$ – текущий символ

$L(\Sigma)$ – язык, рас-мый ДКА

$L(\Sigma) \subseteq \Sigma^*$

$\delta(q_k, \bar{q}_i) \to q_{k+1}$

$\delta : Q \times \Sigma \to Q \setminus q_0$

Пятерка ДКА:

$\langle Q, F, q_0, \Sigma, \delta \rangle$

Диаграммы Мура – ориентированный корневой граф:

- Множество вершин, которое совпадает с Q.
- Каждому ребру соот-ет направление и символ из алфавита.

N 4

Машина Тьюринга — бесконечная лента:

← головка (указатель)

...	0	1	0	1	1	0	0	...

Q — внутренний алфавит — множество конечных состояний

Σ — внешний алфавит — алфавит ленты.

Действия:

- Записать $a \in \Sigma$
- Перейти в $q_k \in Q$
- Перейти по ленте

q_F — конечное состояние.

Тек. лента Тек. сост. Тек. пол. гол.	$\xrightarrow{MT}$	Новая лента Новое состояние Новое пол. гол.

$L = \{L, R, N\}$ — перемещение головки.

$MT(Q, \Sigma, q_F, \delta, L)$, где

$\delta: \Gamma \times Q \setminus \{q_F\} \Rightarrow \Gamma \times Q \times L$

Все, что может быть вычислено, вычислимо на МТ.

N5

- Сложение: стереть пробел и ед-ку
- Вычитание: убрать 2 „1", отдельно 1 и 2.
- Умножение: проверить, что число отрицательно или неот-но; в отр. случ. убрать „1" и перенести сразу; остальные перенести в 1:2; в неотр. 1 перенести вручную в 1:2; остальные в 1:2. Прибавлять одну до, одну после.
- Деление: 4 случая, в каждом свой остаток. Отриц. четное: 1 оставляем, а остальные удаляем причитая 2 удалили, вернулись 1 добавили. Четные: удаляем

3 после 1 „1", две далее. Нечетные: остаток, а далее четные.

N6

Базис Клини:

$O(x)=0$ — константа

$S(x)=x+1$ — следование

$P_n^k(x_1,\dots,x_n)=x_k$ — проектор

Операции:

Суперпозиция: $f(x^m)=h(g^n(x^m))$

Применение рекурсии:

$f(x_1,\dots,x_n)$

$h(x_1,\dots,y+1)=g(x_1,\dots,x_n,y,h(x,\dots,y))$

$h(x_1,\dots,x_m,0)=f(x_1,\dots,x_m)$.

Операция минимизации:

$f(x_1,\dots,x_n,y)=0$

$h(x_1,\dots,x_n)=\min y$.

Рекурсивные функции – функции с неограниченной областью определения

Рекурсивные

Частично рекурсивные (с minim)

Примитивно рекурсивные (1 – 5)

Пример частичной рекурсии:

Функция Аккермана:

$$A(0, x, y) = x + y$$

$$A(z+1, x, 0) = Sg(z)$$

$$A(z+1, x, y+1) = A(z, x, A(z, x, y))$$

$$\min y = Sg(z)$$

$$f(x) = \begin{cases} y - x, & x > 0 \\ \varnothing, & x = 0 \end{cases}$$

$$\min y = x$$

Класс частично рекурсивных функций совпадает с классом вычислимых

Билет 7.

$y \neq 0 \qquad 0 = O(x) \qquad x+1 = S(x)$

$y+1 = S(y)$

1) $Sum(x, 0) = I_3^1(x, 0, Sum) = x$ когда $y=0$

$Sum(x, y+1) = S(Sum(x, y))$

2) $Mul(x, 0) = I_3^2(x, 0, Mul(x,0)) = 0$ когда $y=0$

$Mul(x, y+1) = Sum(x, Mul(x, y))$

3) $Exp(x, 0) = I_3^2(x, S(0), Exp(x,0)) = S(0) = 1$

$Exp(x, y+1) = Mul(x, Exp(x, y))$ когда $y=0$

4) $TETR(x, 0) = S(I_3^2(x, 0, TETR)) = S(0) = 1$

$TETR(x, y+1) = Exp(x, TETR(x, y))$ когда $y=0$

5) $Fac(0) = S(I_2^1(0, Fac(0))) = S(0) = 1$

$Fac(S(0)) = I_2^1(S(0), Fac(0)) = S(0) = 1$

$Fac(x+1) = Mul(x+1, Fac(x))$

6) $$Pred(x) = \begin{cases} x - 1, & x \geq 1 \\ 0, & x < 1 \end{cases}$$

$$Pred(0) = I_2^1(0, Pred(0)) = 0$$

$$Pred(x+1) = \underbrace{S(S(\ldots}_{x \text{ paz}}(Pred(0)))), \quad x \geq 1$$

7) $$\begin{cases} DiFF(x,y) = x - y & (x > y) \\ DiFF(x,y) = 0 & (x \leq y) \end{cases}$$

$$DiFF(x, 0) = I_3^1(x, 0, DiFF(x,0)) = x$$

$$DiFF(x,y) = DiFF(Pred(x), Pred(y))$$

8) $$ABS(0, y) = I_3^2(0, y, ABS(0, y))$$

$$ABS(x, 0) = I_3^1(x, 0, ABS(x, 0))$$

$$ABS(x, y) = ABS(Pred(x), Pred(y))$$

9) $$sg(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

$$sg(S(0)) = I_2^1(S(0), sg(S(0))) = S(0) = 1$$

$$sg(x) = sg(Prev(x)), \quad x > 0$$

10) $antisg = \begin{cases} 0, & x>0 \\ 1, & x \le 0 \end{cases}$

$anti\,Sg(S(0)) = Prev(I_2^1(S(0), antisg(S(0))) = Prev(S(0)) =$
$= 0$

$antisg(x) = antisg(Prev(x)),\ x > 0.$

11) $\underline{x \le y}$

$DiFF(x, y) = 0$

$REM(x, y) = I_3^1(x, y, REM(x, y)) = x$

$\underline{x > y}$

$REM(x, y) = REM(I_3^3(x, y, DIFF(x, y)), y) =$

$= x - \underbrace{y - y - \ldots}_{\boxed{x > y}}$

12) $x < y$

$MOD(x, y) = 0(I_3^1(x, y, MOD(x, y)) =$

$= 0(x) = 0$

$x \geq y$

$\frac{x-y}{y} = \frac{x}{y} - 1 \Rightarrow \frac{x}{y} = 1 + \frac{x-y}{y}$

$MOD(x, y) = S(I_3^3(x, y, MOD(DiFF(x, y), y))$

Билет N 8

Вычислительная сложность – функция зависимости объема работы, которая выполняется некоторым алгоритмом, от размеров входных данных

w – слово – упорядоченное множество символов

$|w|$ – длина слова

Класс P: $t(|w|) = P_N(|w|)$ – арифметика – сортировка – Эйлеров путь

Класс exp: $t(|w|) \leq exp(P_N(|w|))$ –

– перебор всех подмножеств набора из n предметов – $O(2^n)$

Класс NP:

RSA – алгоритм шифрования – можно взломать, зная ключ

NP не решается за полиномиальное время, но проверяется за него

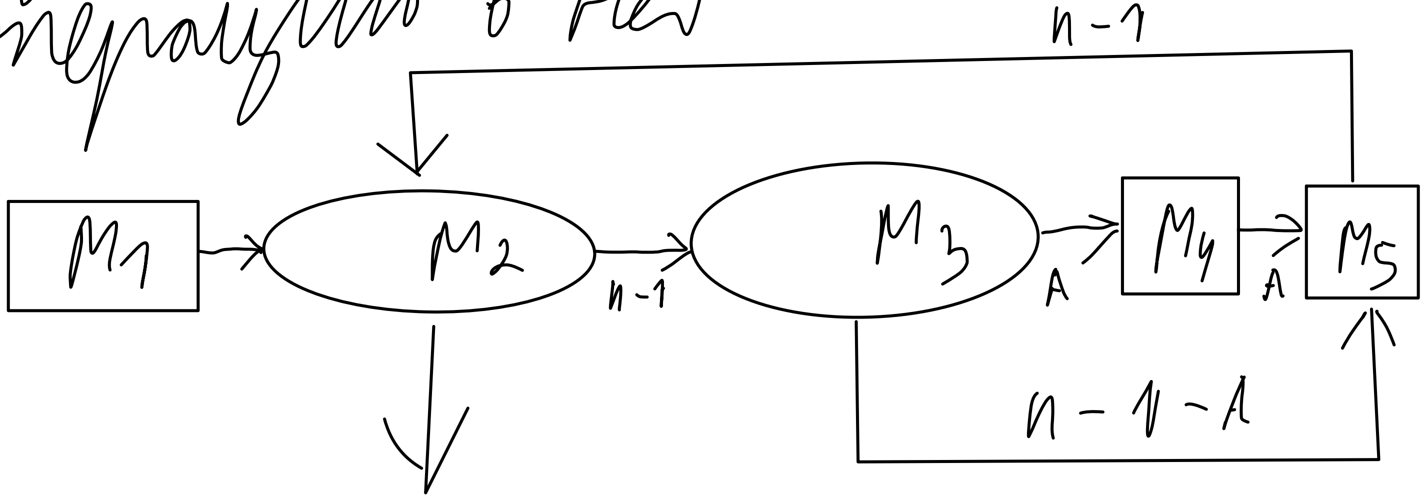
NP-полные задачи – задачи, к которым сводятся NP-задачи.

NP – МЗ решаются за пол-ное время. Ответ: да/нет.

Пример: проверка формулы

Билет № 9.

Количественный анализ эффективности алгоритмов – оценка кол-ва выполнений операций в нем



M_1 – инициализация

M_2 – проверка, что мы обошли массив

M_3 – сравнение

M_4 – присв.

M_5 – уменьшение счетчика

$\min A = 0$

$\max A = n-1$

p_{nk} = (число перестановок n объектов, для которых $A = k$) / $n!$

$$A_n = \sum_k k p_{nk}$$

$G(z) = p_0 + p_1 z + p_2 z^2 + \dots$, где p_k – вероятность того, что некоторая величина принимает значение k.

$$\operatorname{mean}(G) = \sum_k k p_k$$

$$\operatorname{var}(G) = \sum_k k^2 p_k - (\operatorname{mean}(G))^2$$

$G(1) = 1$ – сумма всех возможных вероятностей

$$G'(z) = \sum_k k p_k z^{k-1} \Rightarrow$$

$$\mathrm{mean}(G(z)) = G'(1) = A_n$$

Тогда:

$$\mathrm{Var}(G(z)) = G''(1) + G'(1) - G'(1)^2$$

Чтобы определить поведение A, найдем вероятности p_{nk}. Это легко сделать методом индукции. Согласно (1) нужно подсчитать число перестановок n элементов, для которых $A = k$. Обозначим это число через P_{nk}. Оно равно $P_{nk} = n!\,p_{nk}$.

Рассмотрим перестановки $x_1 x_2 \ldots x_n$ элементов $\{1, 2, \ldots, n\}$ (см. раздел 1.2.5). Если $x_1 = n$, то значение A *на единицу больше*, чем значение для перестановки $x_2 \ldots x_n$. Если же $x_1 \neq n$, то значение A является *точно таким же*, как для перестановки $x_2 \ldots x_n$. Следовательно, $P_{nk} = P_{(n-1)(k-1)} + (n-1)P_{(n-1)k}$, что эквивалентно соотношению

$$p_{nk} = \frac{1}{n}\,p_{(n-1)(k-1)} + \frac{n-1}{n}\,p_{(n-1)k}. \tag{4}$$

* $p = P\{|A - A_n| > 2\sigma_n\}$. — ***Прим. ред.***

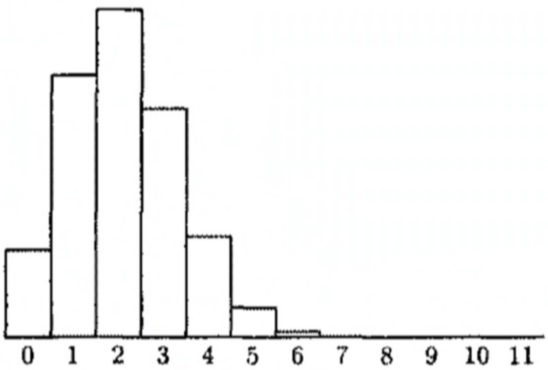


Рис. 10. Распределение вероятностей для шага M4 при $n=12$ Среднее равно 58301/27720, т е приближенно равно 2.10. Дисперсия приближенно равна 1 54

По этой формуле можно найти p_{nk}, если задать начальные условия:

$$p_{1k} = \delta_{0k}; \qquad p_{nk} = 0, \quad \text{если } k < 0 \tag{5}$$

Теперь можно исследовать величины p_{nk} с помощью производящих функций. Пусть

$$G_n(z) = p_{n0} + p_{n1}z + \cdots = \sum_k p_{nk} z^k. \tag{6}$$

Известно, что $A \leq n - 1$, поэтому $p_{nk} = 0$ для больших значений k Таким образом, функция $G_n(z)$ — это на самом деле многочлен, хотя для удобства она и представлена в виде бесконечного ряда

Из (5) получаем, что $G_1(z) = 1$, и согласно (4)

$$G_n(z) = \frac{z}{n}\,G_{n-1}(z) + \frac{n-1}{n}\,G_{n-1}(z) = \frac{z+n-1}{n}\,G_{n-1}(z). \tag{7}$$

В нашем случае нужно вычислить $G'_n(1) = A_n$ Из (7) имеем

$$G'_n(z) = \frac{1}{n}G_{n-1}(z) + \frac{z+n-1}{n}G'_{n-1}(z);$$

$$G'_n(1) = \frac{1}{n} + G'_{n-1}(1).$$

Учитывая начальное условие $G'_1(1) = 0$, находим

$$A_n = G'_n(1) = H_n - 1. \qquad (1\text{‹}$$

Каждый раз, когда необходимо получить информацию о последовательности чисел $\langle a_n \rangle = a_0, a_1, a_2, \ldots$, можно рассмотреть бесконечную сумму от "параметра" z

$$G(z) = a_0 + a_1 z + a_2 z^2 + \cdots = \sum_{n\geq 0} a_n z^n \qquad (1)$$