Токмаков Александр, группа БПМИ165
Домашняя работа 1

№1

В случае, если $m = 0$ возникает неоднозначность: слово 0^p может быть записано как $0^0 1^0 0^p$ (тогда $f(0^p) = f(0^0 1^0 0^p) = 1,\ 0 = 0$) или как $0^q 1^0 0^{p-q},\ 1 \le q \le p$ (тогда $f(0^p) = f(0^q 1^0 0^{p-q}) = 0,\ q \ne 0$). Будем считать, что $f(0^p) = 1$.

Сначала головка МТ движется вправо до конца слова, затем движется обратно, заменяя '0' на '#' (пустой символ) до тех пор, пока не встретит '1' или '#'. Если единиц не встретилось, на ленту записывается ответ '1'. Иначе, головка возвращается к началу слова. Таким образом, на ленте останется слово $0^n 1^m$. (Состояния q_i).

Затем головка МТ движется вправо до первого символа '0' и заменяет его на 'a'. Если '0' не встретился, головка продолжает двигаться вправо до символа '1' (для какой-то единицы не хватило нуля, $n \ne m$, на ленту записывается ответ '0') либо до символа '#' (заменено равное количество единиц и нулей, $n = m$, на ленту записывается ответ '1'). Если '0' встретился, движется вправо до первого символа '1' или '#'. Если встретилась '#', на ленту записывается ответ '0' (для какого-то нуля не нашлось единицы). Иначе головка перемещается на начало слова, повторяются действия, описанные в этом абзаце. (Состояния p_i).

В таблице переходов записаны тройки (новый символ, новое состояние, направление движения); символ '-' в таблице значит, что такая пара (символ, состояние) не может встретиться; начальное состояние q_{rskip}; конечные состояния a (accept) и r (reject).

	$\#$	0	1	a	b
q_{rskip}	$\#, q_{del0}, L$	$0, q_{rskip}, R$	$1, q_{rskip}, R$	$-$	$-$
q_{del0}	$1, a, L$	$\#, q_{del0}, L$	$1, q_{0lskip}, L$	$-$	$-$
q_{lskip}	$\#, p_{repl0}, R$	$0, q_{lskip}, L$	$1, q_{lskip}, L$	$-$	$-$
p_{repl0}	$-$	a, q_{repl1}, R	$1, s_{find1}, N$	a, p_{repl0}, R	b, p_{find1}, N
p_{repl1}	$\#, r_{rskip}, L$	$-$	b, p_{lskip}, L	$-$	b, p_{repl1}, R
p_{lskip}	$\#, p_{repl0}, R$	$0, p_{lskip}, L$	$1, p_{lskip}, L$	a, p_{lskip}, L	b, p_{lskip}, L
p_{find1}	$\#, a_{del}, L$	$-$	$1, r_{rskip}, N$	$-$	b, p_{find1}, R
r_{rskip}	$\#, r_{del}, L$	$0, r_{rskip}, R$	$1, r_{rskip}, R$	a, r_{rskip}, R	b, r_{rskip}, R
r_{del}	$0, r, L$	$\#, r_{del}, L$	$\#, r_{del}, L$	$\#, r_{del}, L$	$\#, r_{del}, L$
a_{del}	$1, a, L$	$\#, a_{del}, L$	$\#, a_{del}, L$	$\#, a_{del}, L$	$\#, a_{del}, L$

№2

Начальное состояние q_{rskip}, конечное состояние q_{stop}

	$\#$	0	1
q_{rskip}	$\#, q_{carry}, L$	$0, q_{rskip}, R$	$1, q_{rskip}, R$
q_{carry}	$1, q_{stop}, L$	$1, q_{lskip}, L$	$1, q_{carry}, L$
q_{lskip}	$\#, q_{stop}, N$	$0, q_{lskip}, L$	$1, q_{lskip}, L$

№3

Обозначения:

$$Z:\ \mathbb{N}\to\mathbb{N},\quad Z(x)=0$$
$$N:\ \mathbb{N}\to\mathbb{N},\quad N(x)=x+1$$
$$U_i^n:\ \mathbb{N}^n\to\mathbb{N},\quad U(x_1,...,x_n)=x_i$$
$$S\langle f,g\rangle:\ \mathbb{N}^m\to\mathbb{N},\quad S\langle f,g\rangle(x_1,...,x_m)=f(g(x_1,...,x_m),...,g(x_1,...,x_m)),\text{ где } f:\ \mathbb{N}^n\to\mathbb{N},\ g:\ \mathbb{N}^m\to\mathbb{N}$$
$$R\langle f,g\rangle:\ \mathbb{N}^{n+1}\to\mathbb{N}\quad R\langle f,g\rangle(x_1,...,x_n,y)=\begin{cases}f(x_1,...,x_n) & if\ y=0\\ g(x_1,...,x_n,y-1,R\langle f,g\rangle(x_1,...,x_n,y-1)) & else\end{cases}$$
$$\text{где } f:\ \mathbb{N}^n\to\mathbb{N},\ g:\ \mathbb{N}^{n+2}\to\mathbb{N}$$

Определим вспомогательные функции (все они примитивно рекурсивные по построению):
Декремент i-того аргумента:

$$Dec(x)=S\langle R\langle Z,U_2^3\rangle,U_1^1,U_1^1\rangle(x)=R\langle Z,U_2^3\rangle(x,x)=\begin{cases}0 & if\ x=0\\ x-1 & else\end{cases}$$
$$Dec_i^n(x_1,...,x_n)=S\langle Dec,U_i^n\rangle(x_1,...,x_n)=Dec(x_i)$$

Вычитание:

$$Sub(x,y)=R\langle U_1^1,Dec_3^3\rangle(x,y)=\begin{cases}0 & if\ x\le y\\ x-y & else\end{cases}$$

Единица от n аргументов:

$$One^n(x_1,...,x_n)=S\langle S\langle N,Z\rangle,U_1^n\rangle(x_1,...,x_n)=N(Z(U_1^n(x_1,...,x_n)))=1$$

Сравнение с нулём:

$$EqZ(x)=R\langle Z,One^2\rangle(x)=\begin{cases}0 & if\ x=0\\ 1 & else\end{cases}$$

Сравнения $x\le y$ и $x\ge y$:

$$Leq(x,y)=S\langle EqZ,Sub\rangle(x,y)=EqZ(x-y)=\begin{cases}0 & if\ x\le y\\ 1 & else\end{cases}$$
$$Geq(x,y)=S\langle Leq,U_2^2,U_1^2\rangle(x,y)=EqZ(y-x)=\begin{cases}0 & if\ x\ge y\\ 1 & else\end{cases}$$

Инкремент i-того аргумента:

$$Inc_i^n(x_1,...,x_n)=S\langle N,U_i^n\rangle(x_1,...,x_n)=N(x_i)=x_i+1$$

Сумма:

$$Sum(x,y)=R\langle U_1^1,Inc_3^3\rangle(x,y)=x+y$$

Сравнение $x=y$:

$$Eq(x,y)=S\langle Sum,Leq,Geq\rangle(x,y)=Leq(x,y)+Geq(x,y)=\begin{cases}0 & if\ x=y\\ 1 & else\end{cases}$$

Поиск значения $f(i)=c$ для $0\le i\le y$:

$$Cmp_f(c,i) = S\langle Eq, U_1^2, S\langle f, U_2^2\rangle\rangle(c,i) = Eq(c, f(i)) = \begin{cases} 0 & if\ f(i) = c \\ 1 & else \end{cases}$$

$$SumCmp_f(c,i,s) = S\langle Sum, S\langle Cmp_f, U_1^3, U_2^3\rangle, U_3^3\rangle(c,i) = Sum(Eq(c, f(i)), s) = \begin{cases} s & if\ f(i) = c \\ s+1 & else \end{cases}$$

$$Find_f(c,y) = R\langle Z, SumCmp_f\rangle(x,y) = |\{i \in \mathbb{N} :\ i \leq y\ \wedge\ f(i) = c\}|$$

Подсчёт пар $(i,j),\ 0 \leq i,j \leq y$, для которых $g(i) = h(i)$:

$$FindVal_{g,h}(j,y) = S\langle Find_g, S\langle h, U_1^2\rangle, U_2^2\rangle(j,y) = Find_g(h(i),y) = |\{i \in \mathbb{N} :\ i \leq y\ \wedge\ g(i) = h(j)\}|$$

$$SumFindVal_{g,h}(y,j,s) = S\langle Sum, S\langle FindVal_{g,h}, U_2^3, U_1^3\rangle, U_3^3\rangle(c,i) =$$

$$= Sum(FindVal_{g,h}(j,y), s) = s + |\{i \in \mathbb{N} :\ i \leq y\ \wedge\ g(i) = h(j)\}|$$

$$FindPairs(x,y) = R\langle Z, SumFindVal_{g,h}\rangle(x,y) = |\{(i,j) \in \mathbb{N}^2 :\ i \leq x\ \wedge\ j \leq y\ \wedge\ g(i) = h(j)\}|$$

$$F(y) = S\langle FindPairs, U_1^1, U_1^1\rangle = FindPairs(y,y) =$$

$$= |\{(i,j) \in \mathbb{N}^2 :\ i,j \leq y\ \wedge\ g(i) = h(j)\}| = \begin{cases} 0 & if\ g(i) \neq h(j)\ for\ 0 \leq i,j \leq y \\ c > 0 & else \end{cases}$$

Осталось сделать так, чтобы $F(y)$ возвращала только 0 или 1:

$$F_1(y) = S\langle EqZ, F\rangle(y) = EqZ(F(y)) = \begin{cases} 0 & if\ g(i) \neq h(j)\ for\ 0 \leq i,j \leq y \\ 1 & else \end{cases}$$

$F_1(y)$ – искомая функция, если h и g примитивно рекурсивны, то F_1 примитивно рекурсивна по построению.

№4

Здесь используются обозначения и некоторые функции из №3.
Определим умножение:

$$Mul(x,y) = R\langle Z, S\langle Sum, U_1^3, U_3^3\rangle\rangle = x \cdot y = \begin{cases} Z(x) & if\ y = 0 \\ Sum(x, Mul(x, y-1)) & else \end{cases}$$

Инвертируем значения Leq:

$$Leq_1(x,y) = S\langle Sub, One^2, Leq\rangle = 1 - Leq(x,y) = \begin{cases} 1 & if\ x \leq y \\ 0 & else \end{cases}$$

Заметим, что $\lfloor \frac{x}{y} \rfloor = z \iff (z+1) \cdot y > x \geq z \cdot y$ и $0 \leq z \leq x$. Таким образом, чтобы найти z, можно перебирать числа от 0 до x до тех пор, пока не начнёт выполняться условие $(z+1) \cdot y > x$:

```
def div(x, y):
    for z in from 0 to x:
        if (z + 1) * y > x:
            return z
    return 0    # let div(x, y) = 0 if y = 0
```

Запишем этот же алгоритм в другом виде:

```
def div(x, y):
    z = 0
    for i in from 0 to x:
        if (z + 1) * y > x:
            z = z
        else
            z = z + 1
    retrun z
```

И ещё раз:

```
def div(x, y):
    z = 0
    for i in from 0 to x:
        z = z + Leq_1((z+1)*y, x)
    retrun z
```

Такой алгоритм уже легко записать с помощью примитивно рекурсивных функций:

$$MulP1^4(x,y,c,z) = S\langle Mul, S\langle N, U_4^4\rangle, U_2^4\rangle(x,y,c,z) = (z+1)\cdot y$$
$$Cond^4(x,y,c,z) = Leq_1(MulP1^4(x,y,c,z),x) = S\langle Leq_1, MulP1^4, U_1^4\rangle(x,y,c,z)$$
$$g(x,y,c,z) = Sum(z, Leq_1(y\cdot(z+1),x)) = S\langle Sum, U_4^4, Cond^4\rangle(x,y,c,z)$$
$$FindZ(x,y,c) = R\langle S\langle Z, U_1^2\rangle g\rangle(x,y,c)$$
$$Div(x,y) = FindZ(x,y,x) = S\langle FindZ, U_1^3, U_2^3, U_1^3\rangle(x,y)$$
$$f(x,y) = S\langle Div, U_1^2, S\langle N, U_2^2\rangle\rangle = Div(x,y+1) = \lfloor \frac{x}{y+1}\rfloor$$

Функция $f(x,y) = \lfloor \frac{x}{y+1}\rfloor$ примитивно рекурсивна.

№5

Для удобства вместо номеров строк будем использовать метки, как в ассемблере (строка, заканчивающаяся двоеточием, указывает на следующую за ней команду). По метке можно легко получить номер команды.
Обозначения:

$T(n,m)$ – скопировать значение из регистра с номером n в регистр с номером m

$Z(n)$ – обнулить регистр с номером n

$S(n)$ – увеличить на 1 значение в регистре с номером n

$J(n,m,\text{LABEL})$ – если значение в регистре с номером n равно значению в регистре с номером m, перейти к строке с меткой LABEL, иначе - к следующей строке

Аргумент вычисляемой функции и вычисленный результат находятся в нулевом регистре.
Программа вычисляет $f(x) = \begin{cases} x+1 & if\ x \mod 2 = 0 \\ x-1 & else \end{cases}$

```
        Z(1)
LOOP1:
        J(0, 1, EVEN)
        T(1, 2)
        S(1)
        J(0, 1, ODD)
        S(1)
        J(0, 0, LOOP1)
EVEN:
        S(0)
        J(0, 0, END)
ODD:
        T(2, 0)
END:
        T(0, 0)
```

№6

Обозначения:

$M(n,m,p)$ – $r_n = r_m \cdot r_p$

Программа вычисляет $f(x,y) = \lfloor \frac{x}{y+1}\rfloor$. Аргумент x передаётся в нулевом регистре, y – в первом регистре, результат – в нулевом регистре.

```
        S(1)
        Z(2)                    // r2 is z = f(x, y)
LOOP:
        T(2, 3)
        S(3)
        M(4, 1, 2)              // r4 = z * (y + 1)
        M(5, 1, 3)              // r5 = (z + 1) * (y + 1)
        // Let's try to find x (r0) in range between r4 and r5
        T(4, 6)
FINDX:
        J(0, 6, FOUND)
        S(6)
        J(5, 6, NOT_FOUND)
        J(0, 0, FINDX)
NOT_FOUND:                      // Let's try whith next z
        S(2)
        J(0, 0, LOOP)
FOUND:          // z * (y + 1) <= x < (z + 1)*(y + 1), so f(x, y) = z
        T(2, 0)
```