Zadanie 1

Przypomnij schematy programów na maszynę licznikową dla podprogramów, rekursji oraz operatora minimalizacji.

Rozwiązanie:

Podprogram:
- kopiowanie argumentów
- zerowanie pozostałych wykorzystywanych rejestrów
- wykonanie podprogramu
- kopiowanie wyliczonego wyniku

Rekursja:
- kopiowanie argumentów funkcji h do bezpiecznego obszaru pamięci
- wyznaczanie wartości funkcji $f(x) = g(x,0)$
- w pętli for wyznaczanie kolejnych wartości $g(x,i+1) = h(x,y,g(x,i))$
- kopiowanie uzyskanego wyniku do rejestru zerowego

Minimalizacja:
- kopiowanie argumentów funkcji f do bezpiecznego obszaru pamięci
- w pętli repeat wyznaczanie kolejnych wartości funkcji $f(x,y)$ z inkrementacją y
- skopiowanie do rejestru zerowego pierwszej odnalezionej wartości y dla której $f(x,y)=0$

Zadanie 2

Napisz program na maszynę licznikową obliczający funkcje:
1. $PLUS(x,y) = x + y$
2. $MNOZ(x,y) = x \cdot y$ *(nie korzystając z PLUS jako procedury)*
3. $MNOZ(x,y) = x \cdot y$ *(z wykorzystaniem PLUS jako procedury)*
4. $MNOZ(x,y) = x \cdot y$ *(z wykorzystaniem schematu rekursji)*

Rozwiązanie:

1. Do rozwiązania części pierwszej zadania wykorzystamy program przykładowy z wykładu:

PLUS(1,2→0)

0	I(2,3,4)
1	S(0)
2	S(3)
3	I(1,1,0)
4	T(1,0)
5	

2. Do zaimplementowania mnożenia wykorzystamy podobny pomysł. Tym razem wykorzystamy dwa rejestry kontrolne – w rejestrze 3 zapiszemy informacje o tym ile razy dodaliśmy już *x*, zaś w rejestrze 4 – ile jedynek dodaliśmy z aktualnie dodawanego *x*-a.

MNOZ(1,2→0)

0	I(2,3,8)
1	I(1,4,5)
2	S(4)
3	S(0)
4	I(1,1,1)
5	Z(4)
6	S(3)
7	I(1,1,0)
8	

3. Wykorzystamy ten sam pomysł używając przygotowanego wcześniej podprogramu PLUS:

MNOZ(1,2→0)

0	I(2,3,4)
1	PLUS(0,1→0)
2	S(3)
3	I(1,1,0)
4	

4. Weźmy $f(x)=0$ oraz $g(x,y,h(x,y))= x+h(x,y)$. Wówczas:

$$h(x,y+1)\begin{cases}0 & \textbf{dla } x=0\\ x+h(x,y) & \textbf{dla } x\neq 0\end{cases}$$

Zadanie 4

Wykaż, że następujące funkcje są ML-obliczalne:

1. $g_1(\overline{x}) = \mu y(f(\overline{x},y) = k),\ k\in \mathbb{N}$,
2. $g_2(\overline{x}) = \mu y(f(\overline{x},y) < k),\ k\in \mathbb{N}$,
3. $g_3(\overline{x}) = \mu y(f(\overline{x},y) \geq k),\ k\in \mathbb{N}$,

Rozwiązanie:

1. Ponieważ chcemy sprawdzić kiedy funkcja przyjmuje wartość *k*, a nie zero, wystarczy zmodyfikować program z dowodu twierdzenia o operatorze minimalizacji dodając na początku *k* razy instrukcję S($m+n+3$). Wówczas, w instrukcji o numerze L+1, gdzie sprawdzany jest warunek końca pętli będziemy porównywać wyznaczoną wartość z *k* a nie z zerem.

	S(n+m+3)
	…
	S(n+m+3)
k	T(1,m+1)
	…
k+n-1	T(n,m+n)
L	F[m+1,…,m+n+1→r]
	I(r,m+n+3,K)
	S(m+n+1)
	I(1,1,L)
K	T(n+m+1,0)

2. Ponieważ chcemy sprawdzić kiedy funkcja przyjmuje wartość mniejszą niż k, a nie zero, wystarczy zmodyfikować program z dowodu twierdzenia o operatorze minimalizacji dodając pętle inkrementującą k-1 razy wartość w $R(m+n+3)$. Wówczas, w instrukcji o numerze "L+1", gdzie sprawdzany jest warunek końca pętli, będziemy porównywać wyznaczoną wartościami od 0 do k-1 a nie z zerem. Należy też pamiętać o zerowaniu wartości w $R(m+n+3)$ przed każdym testem.

	S(n+m+4)	
	…	
	S(n+m+4)	Z[n+m+4]=k
	T(1,m+1)	
	…	
	T(n,m+n)	
L	F[m+1,…,m+n+1→r]	
	Z(m+n+3)	zerowanie
U	I(m+n+4,m+n+3,V)	czy osiągnęliśmy k?
"L+1"	I(r,m+n+3,K)	
	S(m+n+3)	kolejna wartość do testów
	I(1,1,U)	goto U
V	S(m+n+1)	
	I(1,1,L)	
K	T(n+m+1,0)	

3. Ponieważ chcemy sprawdzić kiedy funkcja przyjmuje wartość większą lub równą k, a nie zero, wystarczy zmodyfikować program z dowodu twierdzenia o operatorze minimalizacji dodając pętle inkrementującą k-1 razy wartość w $R(m+n+3)$. Postępować będziemy podobnie jak w przypadku zadania 2, należy tylko odwrócić warunek (równość odpowiednich rejestrów jest teraz warunkiem pozostania w pętli a nie wyjścia z niej). Zauważmy jeszcze, że w oryginalnym dowodzie nie ma różnicy, czy inkrementacji komórki $m+n+1$ dokonamy przed czy po instrukcji testującej. W tym przypadku jest jednak istotne, aby zrobić to przed testem.

	S(n+m+4)	
	…	
	S(n+m+4)	Z[n+m+4]=k
	T(1,m+1)	
	…	
	T(n,m+n)	
L	F[m+1,…,m+n+1→r]	
	S(m+n+1)	przesunięta wcześniej inkrementacja
	Z(m+n+3)	zerowanie
U	I(m+n+4,m+n+3,K)	czy osiągnęliśmy k?
	I(r,m+n+3,L)	
	S(m+n+3)	kolejna wartość do testów
	I(1,1,U)	goto U
K	T(n+m+1,0)	

Zadanie 3

Podaj przykład funkcji totalnej, dla której zastosowanie operatora minimalizacji da w wyniku funkcję, która nie jest totalna.

Rozwiązanie:
Weźmy funkcję f(x,y)=x+y. Wówczas $g(x) = \mu y(f(x,y) = 0)$, określona jest tylko dla $x=0$, gdzie wartość $g(0)=0$, gdyż $f(0,0)=0$. Dla każdej innej wartości x, wartość $f(x,y)>x$, czyli w szczególności $f(x,y)\neq 0$, a więc wartość $g(x)$ jest nieokreślona.

Zadania domowe:

A. Zapisz równoważny instrukcji T(n,m) ciąg złożony z pozostałych instrukcji maszyny licznikowej.

B. Wykaż, że następujące funkcje są ML-obliczalne:

1. $g_1(\overline{x}) = \mu y(f(\overline{x},y) > k)$, $k \in \mathbb{N}$,
2. $g_2(\overline{x}) = \mu y(f(\overline{x},y) \leq k)$, $k \in \mathbb{N}$,

C. Uzasadnij, że jeśli $f \in \mathbb{C}_n$, to również $f \in \mathbb{C}_{n+1}$.