

# Metody programowania 2017

## Zestaw makroinstrukcji

Instrukcja	Pseudokod i opis	
MNot	$D := \text{not } D$	
MNeg	$D := - D$	
MAnd	$D := \text{Memory}[U, U+1] \text{ and } D; U := U + 2$	
MOr	$D := \text{Memory}[U, U+1] \text{ or } D; U := U + 2$	
MAdd	$D := \text{Memory}[U, U+1] + D; U := U + 2$	
MSub	$D := \text{Memory}[U, U+1] - D; U := U + 2$	
MMul	$D := \text{Memory}[U, U+1] * D; U := U + 2$	
MDiv	$D := \text{Memory}[U, U+1] \text{ div } D; U := U + 2$	
MMod	$D := \text{Memory}[U, U+1] \text{ mod } D; U := U + 2$	
MGetLocal $n$	$D := \text{Memory}[U+2n, U+2n+1]$ Pobiera $n$ -te słowo ze stosu do akumulatora	
MSetLocal $n$	$\text{Memory}[U+2n, U+2n+1] := D$ Zapisuje akumulator jako $n$ -te słowo na stosie	
MPush	$U := U - 2; \text{Memory}[U, U+1] := D$ Odkłada akumulator na stos	
MPopN $n$	$U := U + 2n$ Usuwa $n$ słów ze stosu	
MPopAcc	$D := \text{Memory}[U, U+1]; U := U + 2$ Zdejmuje wartość ze stosu i zapisuje ją w akumulatorze	
MBranch $cc \ l$	if $cc$ then $PC := l$ Skocz do etykiety $l$ jeśli spełniony jest warunek $cc$	
MJump $l$	$PC := l$ Skocz do etykiety $l$ (bezwarunkowo)	
MJumpAcc	$PC := D$ Skocz do adresu zapisanego w akumulatorze	
Warunek	Efekty uboczne (pseudokod)	Kiedy spełniony?
MC_1		zawsze
MC_0		nigdy
MC_Z		jeśli $D = 0$
MC_NZ		jeśli $D \neq 0$
MC_P		jeśli $D > 0$
MC_NP		jeśli $D \leq 0$
MC_N		jeśli $D < 0$
MC_NN		jeśli $D \geq 0$
MC_EQ	$\text{tmp} := \text{Memory}[U, U+1]; U := U + 2$	jeśli $\text{tmp} = D$
MC_NE	$\text{tmp} := \text{Memory}[U, U+1]; U := U + 2$	jeśli $\text{tmp} \neq D$
MC_LT	$\text{tmp} := \text{Memory}[U, U+1]; U := U + 2$	jeśli $\text{tmp} < D$
MC_GE	$\text{tmp} := \text{Memory}[U, U+1]; U := U + 2$	jeśli $\text{tmp} \geq D$
MC_LE	$\text{tmp} := \text{Memory}[U, U+1]; U := U + 2$	jeśli $\text{tmp} \leq D$
MC_GT	$\text{tmp} := \text{Memory}[U, U+1]; U := U + 2$	jeśli $\text{tmp} > D$

Instrukcja	Pseudokod i opis
MCall $l$	$S := S - 2$ ; $\text{Memory}[S, S+1] := PC$ ; $PC := l$ Zawołaj procedurę znajdującą się pod etykietą $l$
MCallAcc	$S := S - 2$ ; $\text{Memory}[S, S+1] := PC$ ; $PC := D$ Zawołaj procedurę, której adres znajduje się w akumulatorze
MRet	$PC := \text{Memory}[S, S+1]$ ; $S := S + 2$ Powrót z procedury
MConst $n$	$D := n$ Załaduj stałą $n$ do akumulatora
MGetLabel $l$	$D := l$ Załaduj adres etykiety $l$ do akumulatora
MAlloc $n$	$D := \text{alloc}(n)$ Przydziel pamięć dla rekordu składającego się z $n$ słów 16-bitowych
MGet $n$	$X := D$ ; $D := \text{Memory}[X+2n, X+2n+1]$ Weź $n$ -te pole rekordu znajdującego się w akumulatorze
MSet $n$	$X := \text{Memory}[U, U+1]$ ; $\text{Memory}[X+2n, X+2n+1] := D$ Zapisz akumulator jako $n$ -pole rekordu znajdującego się na stosie