|  |  |
| --- | --- |
| load x | kopiert den Wert aus der Speicherzelle x in den Akkumulator |
| loadi n | lädt die ganze Zahl n in den Akkumulator |
| store x | kopiert den Wert aus dem Akkumulator in die Speicherzelle x |
| add x | addiert den Wert in Speicherzelle x zum Wert im Akkumulator |
| addi n | addiert die ganze Zahl n zum Wert im Akkumulator |
| sub x | subtrahiert den Wert in Speicherzelle x vom Wert im Akkumulator |
| subi n | subtrahiert die ganze Zahl n vom Wert im Akkumulator |
| div x | dividiert den Wert im Akkumulator durch den Wert in Speicherzelle x (ganzzahlige Division) |
| divi n | dividiert den Wert im Akkumulator durch die ganze Zahl n (ganzzahlige Division) |
| mod x | speichert den Rest bei der Ganzzahldivision des Akkumulators durch den Wert in Speicherzelle x in den Akkumulator |
| modi n | speichert den Rest bei der Ganzzahldivision des Akkumulators durch die ganze Zahl n in den Akkumulator |
| jmp x | springt zum Befehl in Speicherzelle x |
| jeq x | springt zum Befehl in Speicherzelle x, falls der Wert im Akkumulator gleich null ist |
| jne x | springt zum Befehl in Speicherzelle x, falls der Wert im Akkumulator ungleich null ist |
| jge x | springt zum Befehl in Speicherzelle x, falls der Wert im Akkumulator gleich null oder positiv ist |
| jle x | springt zum Befehl in Speicherzelle x, falls der Wert im Akkumulator gleich null oder negativ ist |
| jgt x | springt zum Befehl in Speicherzelle x, falls der Wert im Akkumulator positiv ist |
| jlt x | springt zum Befehl in Speicherzelle x, falls der Wert im Akkumulator negativ ist |
| hold | beendet die Abarbeitung des Programms |