**PUSH xy** – iš atminities į steką įkeliami duomenys   
 [SP] = [x \* blank + y]; SP = SP + 1

**PSHC x** – į steką įkeliama konstanta  
 [SP] = x; SP = SP + 1

**POP xy** – iš steko į atmintį įkeliami duomenys  
 SP = SP – 1; [x \* blank + y] = [SP]

**POP** – iš steko ištrinami duomenys  
 SP = SP – 1

**TOP xy** – steko viršūnės reikšmė įkeliama į atmintį  
 [x \* blank + y] = [SP – 1]

**ADD** – sudedamos steko viršūnėje esančios reikšmės  
 [SP – 2] = [SP – 2] + [SP – 1]; SP = SP – 1

**SUB** – atimamos steko viršūnėje esančios reikšmės  
 [SP – 2] = [SP – 2] - [SP – 1]; SP = SP – 1

**MULT** – sudauginamos steko viršūnėje esančios reikšmės  
 [SP – 2] = [SP – 2] \* [SP – 1]; SP = SP – 1

**DIV** – padalinamos steko viršūnėje esančios reikšmės  
 [SP – 2] = [SP – 2] / [SP – 1]; SP = SP – 1

**JZ xy** – valdymas perduodamas, jei loginio trigerio reikšmė yra „F“  
 IF([C] = “F”) PC = x \* blank + y;

**//ALTERNATIVE\\  
JZ xy** – valdymas perduodamas, jei steko viršūnėje yra 0  
 SP = SP – 1; IF([SP] == 0) PC = x \* blank + y;

**//\*\*\*\*\*\*\*\*\*\*\*\*\\**

**JMP xy** – valdymas perduodamas  
 PC = x \* blank + y; SP = SP – 1

**HALT** – programos vygdymo pabaiga

**// BONUS \\  
GRT** – palyginami steko elementai  
 IF([SP – 2 ] > [SP – 1]) [SP – 2] = 1;  
 ELSE [SP – 2] = 0;  
 SP = SP – 1

**LESS** – palyginami steko elementai  
 IF([SP – 2 ] < [SP – 1]) [SP – 2] = 1;  
 ELSE [SP – 2] = 0;  
 SP = SP – 1

**EQ** – palinami steko elementai  
 IF([SP – 2 ] == [SP – 1]) [SP – 2] = 1;  
 ELSE [SP – 2] = 0;  
 SP = SP – 1

**NOT** – steko viršūnei priskiriama priešinga loginė reikšmė  
 IF([SP – 1] == 0) [SP – 1] = 1;  
 ELSE [SP – 1] = 0

**OR** – tikrinamos dvi steko viršūnės reikšmės  
 IF([SP – 1] == 1 || [SP – 2] == 1) [SP – 2] = 1;  
 ELSE [SP – 2] = 0;  
 SP = SP – 1

**AND** - tikrinamos dvi steko viršūnės reikšmės  
 IF([SP – 1] == 1 && [SP – 2] == 1) [SP – 2] = 1;  
 ELSE [SP – 2] = 0;  
 SP = SP – 1