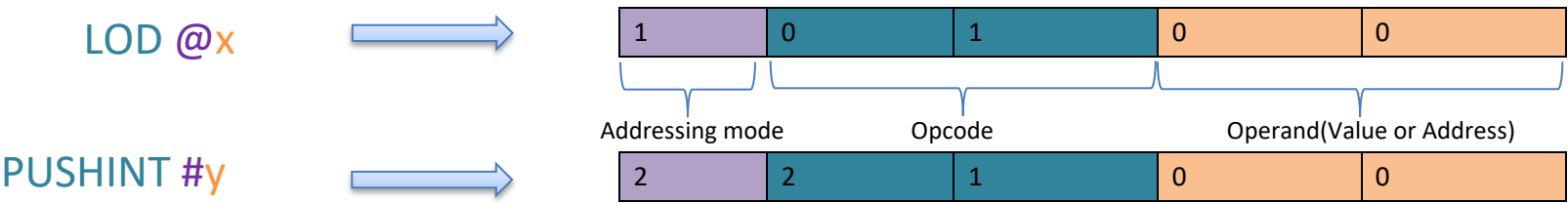


GeonSSembly Instruction Set Table



Instruction	Overall RT	Address Mode (operand 맨 앞 2비트) 0 = 값 or 피연산자없음 1 = 전역변수(DS) 2 = 지역변수(SS) 3 = 멤버변수(HS)	Opcode	Meaning
HALT	Exit	0	0x00	프로그램 종료
LOD (x)	AC<-MBR<-(x)	0,1,2,3	0x01	주소(address mode = 1, 2, 3)일 경우: 현재 변수(피연산자)의 주소에 해당하는 값을 MBR, AC에 적재 값(address mode = 0)일 경우 : 명시된 값(피연산자)을 MBR, AC에 적재
STO (x)	(x)<-MBR<-AC	1,2,3	0x03	x주소에 현재 AC값 저장
ADD (x)	AC<-AC+(x)	0,1,2,3	0x05	현재 AC의 값에 x를 더하고 그 값을 AC값으로 업데이트
SUB (x)	AC<-AC-(x)	0,1,2,3	0x06	현재 AC의 값에 x를 빼고 그 값을 AC값으로 업데이트
MUL (x)	AC<-AC*(x)	0,1,2,3	0x07	현재 AC의 값에 x를 곱하고 그 값을 AC값으로 업데이트
DIV (x)	AC<-AC/(x)	0,1,2,3	0x08	현재 AC의 값에 x를 나누고 그 값을 AC값으로 업데이트
EQ (x)	AC <- AC&&(x)	0,1,2,3	0x0a	현재 AC의 값과 x를 ==연산 (참이면 0)(거짓이면 1)을 AC값으로 업데이트
NOT (x)	AC<-!(AC==(x))	0,1,2,3	0x0b	현재 AC의 값과 x를 !=연산 (참이면 0)(거짓이면 1)을 AC값으로 업데이트
JMP x	PC<-x	0	0x10	현재 PC의 값을 x로 변경
GZJ x	If(AC>0) then PC<-X	0	0x11	현재 AC값이 0보다 크다면 현재 PC값을 x로 변경
BZJ x	if(AC<0) then PC<-X	0	0x12	현재 AC값이 0보다 작다면 현재 PC값을 x로 변경
EZJ x	if(AC==0) then PC <-x	0	0x13	현재 AC값이 0과 같다면 현재 PC값을 x로 변경
DL (x)	1.(x)<-SP 2.Memory[SS+(x)]<-(x) 3.SP += 4	2	0x20	(Dinamic Link with integer Local Variable) 현재 Stack Pointer의 값을 x에 저장한 뒤 x를 StackSegment의 마지막에 저장하고 StackPointer += 4로 업데이트 e.g.,)x가 4, SS의 절대주소가 Memory[3000]부터이고, 스택에 쌓인 것이 없어 SP가 0이라면 x = 0으로 변경, Memory[3000+0]에 저장, StackPointer 4증가 (current StackPointer = 4)
PUSHINT (x)	1.Memory[SS+(x)]<-(x) 2.SP += 4	2	0x21	(Push integer Local Variable into Stack Segment) x를 Stack Segment의 마지막에 저장하고 StackPointer 값을 정수형 변수의 크기인 4바이트만큼 증가시킴 e.g.,)x가 8, SS의 절대주소가 Memory[3000]부터이면 Memory[3000+8]에 저장, stackPointer 4증가 (current StackPointer = 8)
STF (x)	1.Memory[SS+(x)] = 0 Memory[SS+(x)+1] = 0 . . . Memory[SS+SP] = 0 2. SP = (x)	2	0x22	(Stack Free from x to SP) x인덱스부터 ~ SP(현재 스택의 마지막)까지 비우고, SP의 값을 x로 변경
NEWINT (x)	Memory[HS+object+(x)]<-(x)	3	0x30	(New Member Variable into Heap Segment) 객체의 정수형 멤버변수 x를 객체의 주소값+x의 주소값하여 힙 세그먼트에 저장 e.g.,)x의 주소값이 8이고 Data Segment에 object type의 obj라는 변수의 주소가 4이면 Memory[HS+4+8]에 저장
PRT (x)	print (x)	0,1,2,3	0x50	x를 출력
INP	AC<-input	0	0x60	입력받고, 입력받은 값을 AC에 적재