SP ass 2:

Roll no: 38 Name: Yash Oswal

1. ALP:

```
     START 500
X  DC 65
     MOVER AREG, Y
     MOVEM BREG, ='11'
     ADD AREG, ='24'
     LTORG
     SUB BREG, ='15'
Y  DC 20
     ORIGIN 800
     LTORG
     BC DOWN
     MOVER AREG, NUM
DOWN  MOVER CREG, NUM
     ADD BREG, ='17'
NUM DS 5
END
```

2. Symbol Table Literal Table Pool Table

```
X    500
Y    511
DOWN 804
NUM  808
```

```
='11'     507
='24'     508
='15'     800
='17'     813
```

```
0
2
3
```

3. Intermediate Code:

```
     (AD,01)   (C,500)
500  (DL,02)   (C,65)
501  (IS,04)   (RG,1)(S,1)
503  (IS,05)   (RG,2)(L,1)
505  (IS,01)   (RG,1)(L,2)
507  (AD,05)   (DL,02)(C,1)
508  (AD,05)   (DL,02)(C,2)
509  (IS,02)   (RG,2)(L,3)
511  (DL,02)   (C,20)
512  (AD,03)   (C,800)
800  (AD,05)   (DL,02)(C,1)
801  (IS,07)   (S,2)
802  (IS,04)   (RG,1)(S,3)
804  (IS,04)   (RG,3)(S,3)
806  (IS,01)   (RG,2)(L,4)
808  (DL,01)   (C,5)
813  (AD,02)
```

4. Object Code:

```
500  (04) (001) (510)
502  (05) (002) (506)
504  (01) (001) (507)
506  (05) (000) (001)
507  (05) (000) (002)
508  (02) (002) (800)
511  (03) (800)
800  (05) (000) (001)
801  (07) (804)
802  (04) (001) (808)
804  (04) (003) (808)
806  (01) (002) (813)
```

5. Source Code:

```
from io import TextIOWrapper

MOT={
      'STOP':('00','IS',0),
      'ADD':('01','IS',2),
      'SUB':('02','IS',2),
      'MUL':('03','IS',2),
      'MOVER':('04','IS',2),
      'MOVEM':('05','IS',2),
      'COMP':('06','IS',2),
      'BC':('07','IS',1),
      'DIV':('08','IS',2),
      'READ':('09','IS',1),
      'DEC':('11','IS',1),
      'START':('01','AD',1),
      'END':('AD',0),
      'ORIGIN':('03','AD',1),
      'EQU':('04','AD',2),
      'LTORG':('05','AD',0),
      'DS':('01','DL',1),
      'DC':('02','DL',1)
}

REG={
      'AREG':1,
      'BREG':2,
      'CREG':3,
      'DREG':4
```

```
}

class files(object):
      ifp=open("tables/inter_code.txt",mode="r")
      lit=open("tables/literal_table.txt","r")
      litline = lit.read().splitlines()
      sym=open("tables/symbol_table.txt","r")
      symline = sym.read().splitlines()
      output=open("tables/output.txt","a+")
      output.truncate(0)

def getInst(s:str):
      i = s.removeprefix('(').removesuffix(')').split(sep=',')[-1]
      return "("+i+")\t"

def getS(s:str):
      a = s.split(sep=',')[-1]
      a = files.symline[int(a)].split(sep='\t')[-1]
      return a

def getL(s:str):
      a = s.split(sep=',')[-1]
      a = files.litline[int(a)-1].split(sep='\t')[-1]
      return a

def getC(s:str):
      a = s.split(sep=',')[-1]
      if len(a)<3:
            if len(a)<2:
                  a = " (00"+a+")"
            else:
                  a = " (0"+a+")"
      return a

def getRest(s:str):
      a='(000)'
      s=s.replace(')(',' ').removeprefix('(').removesuffix(')').split()
      if 'RG' in s[0]:
            a = "(00"+s[0].split(',')[-1]+")"
      elif 'DL' in s[0]:
            a = "(000)"
      elif 'S' in s[0]:
            a = "("+getS(s[0])+")"
      else:
            a = s[0].split(',')[-1]
            if len(a)<3:
                  if len(a)<2:
                        a = "(00"+a+")"
                  else:
                        a = "(0"+a+")"
            else :
                  a = "("+a+")"
```

```
	if len(s) > 1:
		if 'S' in s[1]:
			a += " ("+getS(s[1])+")"
		if 'L' in s[1]:
			a += " ("+getL(s[1])+")"
		if 'C' in s[1]:
			a += getC(s[1])
	return a

def pass_two(intermediateCode: TextIOWrapper):
	objctCode=''
	i=''
	r=''
	l=0
	for line in intermediateCode:
		l+=1
		if "(AD,01)" in line or "(AD,02)" in line:
			pass
		elif "DL" in line.split()[1]:
			pass
		else:
			words = line.split()
			i=getInst(words[1])
			r=getRest(words[2])
			objctCode += (words[0]+"\t"+i+r+"\n")
	files.output.writelines(objctCode)
	return

##Program execution starts here
pass_two(intermediateCode=files.ifp)
```