ADD A NEW INSTRUCTION:-

If you want to add a new instruction in the emulator you have to make some changes in header files.

You need not make any changes in MAINFILE.C.

Directly go to Execution.h file and insert case of that instruction and define condition for it and call a function that will perform that operation. This function should be defined in anyone of the header files where it is most appropriate.

Let's say if we want to insert an instruction SLT (it is a compare instruction) that is of R-type instruction format that means it has two source register rs1 and rs2 and one destination register rd.

Makes its condition as

```
case 's': if(inst[i+1]=='l'&&inst[i+2]=='t'&&(inst[i+3]==' '||inst[i+3]=='/t')

{

    i=i+4;

    SLT(inst,i);  //call a function which performs set if less than operation

}
```

Here, inst is a character array that stores current instruction.

Now, you have to define a function in any of another header file or you may create your own header file but you must include it.

```
void SLT(char *inst, int i)

{

  Rtype(inst, i);

If(xreg[x1]<xreg[x2])

xreg[xd]=1;
```

else

xreg[xd]=0;

}

Now, define your function in your file. In this process you will need identify source and destination registers or immediate(if present).

There are some function (Rtype(), Itype(), ItypeL(), Btype(), Utype(), Stype()) in Register.h header file that will help you if you have similar instruction format or you may define it by own.

Instruction format	Function
op rd, rs1, rs2	Rtype()
op rd, rs1, imm	Itype()
op rd, imm	Utype()
op rs1, rs2, imm/label	Btype()
op rd, imm(rs1)	ItypeL()
op rs1, imm(rs2)	Stype()

Choose a function according to instruction format.

It takes string in which instruction is stored and i(index : where pointing in string) as argument and specify its index if register and its value if immediate.

Now, write code in that function(SLT()) to perform appropriate operation.