

```

import java.lang.reflect.Array;

public class DataCycle {

    //Properties
    Instruction_ instruction = new Instruction_();
    int[] mainMem = new int[1024];
    public static int[] regs = new int[32];

    int clockCycle = 0;
    Instruction_[] setOfInstructionObjects = new Instruction_[11];
    String regString = "";

    //Stages
    IF_ID theIfIdStage = new IF_ID();
    ID_EX theIdExStage = new ID_EX();
    EX_MEM theExMemStage = new EX_MEM();
    MEM_WB theMemWbStage = new MEM_WB();

    //Set up instruction environment
    int[] instructions = { 0xa1020000, 0x810AFFFC, 0x00831820, 0x01263820, 0x01224820, 0x81180000,
0x81510010,
                        0x00624022, 0x00000000, 0x00000000, 0x00000000};

    //Constructors
    public DataCycle(){

    }

    //The Program
    public void run( int numberOfCycles ){

        //Set up memory
        for (int i = 0; i < 1024; i++) {
            mainMem[i] = i & 255;
        }
        regs[0] = 0;
        for (int i = 1; i < 32; i++) {
            regs[i] = i + 100;
        }

        //turn instruction to objects for disassembler
        int instructionsLen = setOfInstructionObjects.length;
        for( int i = 0; i < instructionsLen; i++ ){
            Instruction_ instObj = new Instruction_();
            instObj.setInstruction(instructions[i]);
            setOfInstructionObjects[i] = instObj;
        }

        Print_out_everything();

        //Run Stages
        for( int i = 0; i < instructionsLen; i++ ){
            IF_stage();
            ID_stage();
            EX_stage();
            MEM_stage();
            WB_stage();
        }
    }
}

```

```

        Print_out_everything();

        Copy_write_to_read();
    }
}

```

```

//Stage Methods
public void IF_stage(){
    this.theIfIdStage.runThisStageWrite(this);
}
public void ID_stage(){
    this.theIdExStage.runThisStageWrite(this);
}
public void EX_stage(){
    this.theExMemStage.runThisStageWrite( this, theIdExStage);
}
public void MEM_stage(){
    this.theMemWbStage.runThisStageWrite(this, theExMemStage);
}
public void WB_stage(){
    this.theMemWbStage.runThisStageWB(this);
}
}

```

```

public void Print_out_everything(){
    System.out.println(this.printCycle());
    clockCycle++;
    regString = "";
    for( int i = 0; i < 32; i++){
        regString += Integer.toString(regs[i]) + " ";
    }
}

```

```

public void Copy_write_to_read(){
    this.theIfIdStage.runThisStageRead();
    this.theIdExStage.runThisStageRead();
    this.theExMemStage.runThisStageRead();
    this.theMemWbStage.runThisStageRead();
}

```

```

//Methods

```

```

public String printCycle(){
    String rtn = "Clock Cycle " + this.getClockCycle() + " \n" ;

    if( this.clockCycle == 0 ){
        rtn += "\nIF/ID: Write \n" + " Inst = " + this.theIfIdStage.inst_write + " IncrPC = " + theIfIdStage.getIncrPC_write();
    }
    else if( this.clockCycle != 0 ){
        if( this.theIdExStage.instruction_write.getType() == "I" ){
            rtn += "\nIF/ID: Write \n" + " Inst = " + this.theIfIdStage.inst_write + " IncrPC = " + theIfIdStage.getIncrPC_write();
        } else if( this.theIdExStage.instruction_write.getType() == "R" ){
            rtn += "\nIF/ID: Write \n" + " Inst = " + this.theIfIdStage.inst_write + " IncrPC = " + theIfIdStage.getIncrPC_write();
        }
    }
}

```

```

        rtn += "\nIF/ID: Read \n";
        if( this.theIfIdStage.inst_read == "0" ){
            rtn += " Inst = " + this.theIfIdStage.inst_read + "      [ NOP ]      " + "IncrPC = "
+ theIfIdStage.getIncrPC_write() + "\n" ;
        }
        else if( this.theIfIdStage.inst_read != "0" ){
            if( this.theIfIdStage.instruction_read.getType() == "I" ){
                rtn += " Inst = " + this.theIfIdStage.inst_read + "      [ " +
this.theIfIdStage.instruction_read.getType() + " " +
this.theIfIdStage.instruction_read.getOpCodeString() + " $" + this.theIfIdStage.instruction_read.rs + ",
" + this.theIfIdStage.instruction_read.immediate + " ($" + this.theIfIdStage.instruction_read.getRt() +
") " + " ]      " + "IncrPC = " + theIfIdStage.getIncrPC_write() + "\n" ;
            } else if( this.theExMemStage.instruction.getType() == "R" ){
                rtn += " Inst = " + this.theIfIdStage.inst_read + "      [ " +
this.theIfIdStage.instruction_read.getType() + " " + this.theIfIdStage.instruction_read.getFuncString()
+ " $" + this.theIfIdStage.instruction_read.rd + ", $" + this.theIfIdStage.instruction_read.rt + ", $" +
this.theIfIdStage.instruction_read.rs + " ]      " + "IncrPC = " + theIfIdStage.getIncrPC_write() + "\n" ;
            }
        }
    }

    rtn += "\nID/EX Write \n";
    rtn += " Control: RegDst = $" + this.theIdExStage.getRegDst_write() + ",    ALUSrc
= " + Integer.toString( this.theIdExStage.getAluSrc_write() ) + ",    ALUOp = " +
this.theIdExStage.aluOp_write + "    MemRead = " + Integer.toString(
this.theIdExStage.getMemRead_write() ) + "\n" ;
    rtn += "    " + "ReadReg1Value = " + this.theIdExStage.getReadReg1Value_write() + "
" + "ReadReg2Value = " + this.theIdExStage.getReadReg2Value_write() + "\n" ;
    rtn += " SEOFFSET = " + this.theIdExStage.getSeoOffset_write() + "    " +
"WriteReg_20_16 = $" + this.theIdExStage.getWriteRef_20_16_write() + "    " + "WriteReg_15_11 = $" +
this.theIdExStage.getWriteReg_15_11_write() + "    " + "Function = " +
this.theIdExStage.getFunction_write() + "\n" ;

    rtn += "\nID/EX Read \n";
    rtn += " Control: RegDst = $" + this.theIdExStage.getRegDst_read() + ",    ALUSrc =
" + Integer.toString( this.theIdExStage.getAluSrc_read() ) + ",    ALUOp = " + "MemRead = " +
Integer.toString( this.theIdExStage.getMemRead_read() ) + "\n" ;
    rtn += "    " + "ReadReg1Value = " + this.theIdExStage.getReadReg1Value_read() + "
" + "ReadReg2Value = " + this.theIdExStage.getReadReg2Value_read() + "\n" ;
    rtn += " SEOFFSET = " + this.theIdExStage.getSeoOffset_read() + "    " +
"WriteReg_20_16 = $" + this.theIdExStage.getWriteRef_20_16_read() + "    " + "WriteReg_15_11 = $" +
this.theIdExStage.getWriteReg_15_11_read() + "    " + "Function = " + this.theIdExStage.getFunction_read() +
"\n" ;

    rtn += "\nEX/MEM Write \n";
    rtn += " Control: MemRead = " + this.theExMemStage.getMemRead_write()+ ",
MemWrite = " + Integer.toString( this.theExMemStage.getMemWrite_write() ) + ",    Branch = " +
this.theExMemStage.getBranch_write() + ",    MemToReg = " + this.theExMemStage.getMemToReg_write() + ",
RegWrite = " + this.theExMemStage.getRegWrite_write() + "\n" ;
    rtn += " CalcBTA = " + this.theExMemStage.getCalcBTA_write() + "    Zero = " +
this.theExMemStage.getZero_write() + "\n" ;
    rtn += " ALUResult = " + this.theExMemStage.getAluResult_write() + "    SWValue = "
+ this.theExMemStage.swValue_write + "    WriteRegNum = $" + this.theExMemStage.getWriteRegNum_write() +
"\n";

    rtn += "\nEX/MEM Read \n";
    rtn += " Control: MemRead = " + this.theExMemStage.getMemRead_read()+ ",
MemWrite = " + Integer.toString( this.theExMemStage.getMemWrite_read() ) + ",    Branch = " +
this.theExMemStage.getBranch_read() + ",    MemToReg = " + this.theExMemStage.getMemToReg_read() + ",
RegWrite = " + this.theExMemStage.getRegWrite_read() + "\n" ;
    rtn += " CalcBTA = " + this.theExMemStage.getCalcBTA_read() + "    Zero = " +
this.theExMemStage.getZero_read() + "    ALUResult = " + this.theExMemStage.getAluResult_read() + "\n" ;

```

```

        rtn += " SWValue = " + this.theExMemStage.getSwValue_read() + "    WriteRegNum = $"
+ this.theExMemStage.getWriteRegNum_read() + "\n";

        rtn += "\nMEM/WB Write \n";
        rtn += " Control:    MemToReg = " + this.theMemWbStage.getMemToReg_write() + ",
RegWrite = $" + this.theMemWbStage.getRegWrite_write() + "\n" ;
        rtn += " LWDDataValue = " + this.theMemWbStage.getLwDataValue_write() + "
ALUResult = " + this.theMemWbStage.getAluResult_write() + "    WriteRegNum = $" +
this.theMemWbStage.getWriteRegNum_write() + "\n";

        rtn += "\nMEM/WB Read \n";
        rtn += " Control:    MemToReg = " + this.theMemWbStage.getMemToReg_read() + ",
RegWrite = $" + this.theMemWbStage.getRegWrite_read() + "\n" ;
        rtn += " LWDDataValue = " + this.theMemWbStage.getLwDataValue_read() + "
ALUResult = " + this.theMemWbStage.getAluResult_read() + "    WriteRegNum = $" +
this.theMemWbStage.getWriteRegNum_read() + "\n";

        rtn += "";
        rtn += "";
        rtn += "\n \n \n";

        rtn += "Regs\n";
        rtn += regString;
        rtn += "\n \n \n";
        return rtn;
    }

    public String toString(){
        String rtn = "Clock Cycle " + this.getClockCycle() + " \n" ;
        rtn += "\nIF/ID: \n" + "Inst = " + this.theIfIdStage.getInst_write() + "    [ " +
this.theIfIdStage.instruction_write.getType() + " ]    " + "IncrPC = " + "PUT INCRPC HERE\n" ;
        rtn += "\nIF/ID \n";
        rtn += "Inst = " + this.theIfIdStage.getInst_read() + "    [ " +
this.theIfIdStage.instruction_write.getInstruction() + " ]" + "IncrPC = " + "PUT INCRPC HERE \n" ;
        rtn += "\nID/EX Write \n";
        rtn += "Control: RegDst = " + this.theIdExStage.getRegDst_write() + ",    ALUSrc = " +
Integer.toString( this.theIdExStage.getAluSrc_write() ) + ",    ALUOp = " + "MemRead = " +
Integer.toString( this.theIdExStage.getMemRead_write() ) ;
        return rtn;
    }

    public Instruction_ getInstructionInSetOfInstructionsObjects(int i){
        return this.setOfInstructionObjects[i];
    }

    public String getInstructionsAtIndex( int i ){
        return Integer.toHexString( this.instructions[i] );
    }

    public EX_MEM getTheExMemStage() {
        return theExMemStage;
    }

    public void setTheExMemStage(EX_MEM theExMemStage) {
        this.theExMemStage = theExMemStage;
    }

    public ID_EX getTheIdExStage() {

```

```
        return theIdExStage;
    }

    public void setTheIdExStage(ID_EX theIdExStage) {
        this.theIdExStage = theIdExStage;
    }

    public IF_ID getTheIfIdStage() {
        return theIfIdStage;
    }

    public void setTheIfIdStage(IF_ID theIfIdStage) {
        this.theIfIdStage = theIfIdStage;
    }

    public MEM_WB getTheMemWbStage() {
        return theMemWbStage;
    }

    public void setTheMemWbStage(MEM_WB theMemWbStage) {
        this.theMemWbStage = theMemWbStage;
    }
    public Instruction_ getInstruction() {
        return instruction;
    }
    public void setInstruction(Instruction_ instruction) {
        this.instruction = instruction;
    }

    public int getClockCycle() {
        return clockCycle;
    }

}
```