Microprocessor Tomasulo Algorithm :

1-Team members:

       1-Mahmoud Mostafa Basha 40-1748

       2- Mohamed gamal Yousef 40-12812

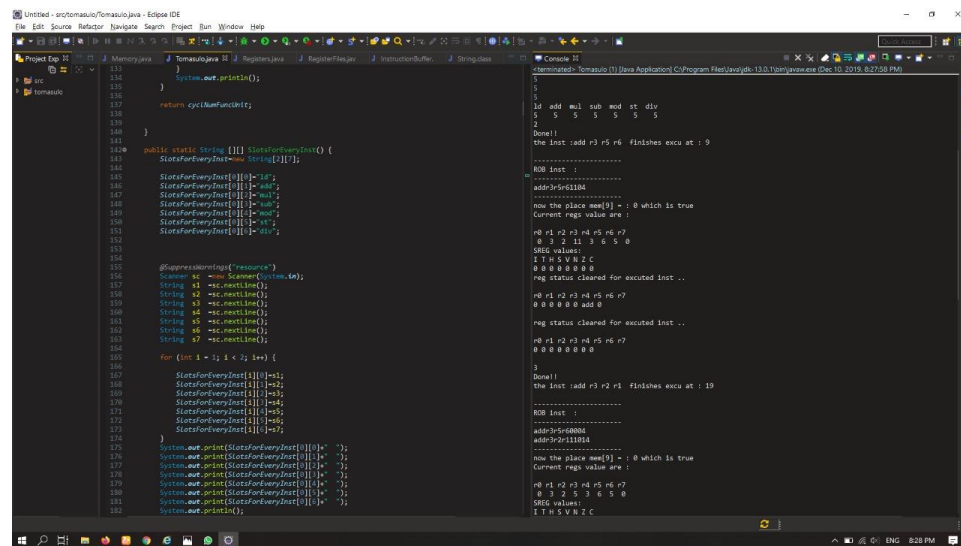   3- Abdelrahman Adel abudif 40-1979

       4-Abdelrahman Elsayed 40-11916

2-First of all we start creating the data structure we are going to use (instructionBuffers ,ReservationStations,RegisterFiles ,Memory,etc..), after that implemented some thodes to write a txt file in the memory and getting that data of the memory into the instructionbuffer splitted to be able to use it easly ,and then start running the fill (fetch) method that takes every single instruction seperately and apply some conditions on it like (availble slot,number of cycles ,register status availability, etc..) to see if its fits the conditions and to asure that we can start filling the stations that we created ,Calling the excution function is the next step is compares the current program clock and see how many clocks we still need to finish the excution ,after indicating that the excution ends we fill the reodered buffer in start commiting in oreder , in the commit statge we start actualy writing the data in the memory and the regs .

3- we will discuss this part in the evaluation

4-the code prints a steps on what to do exactly in each step ,here are the snap shots :
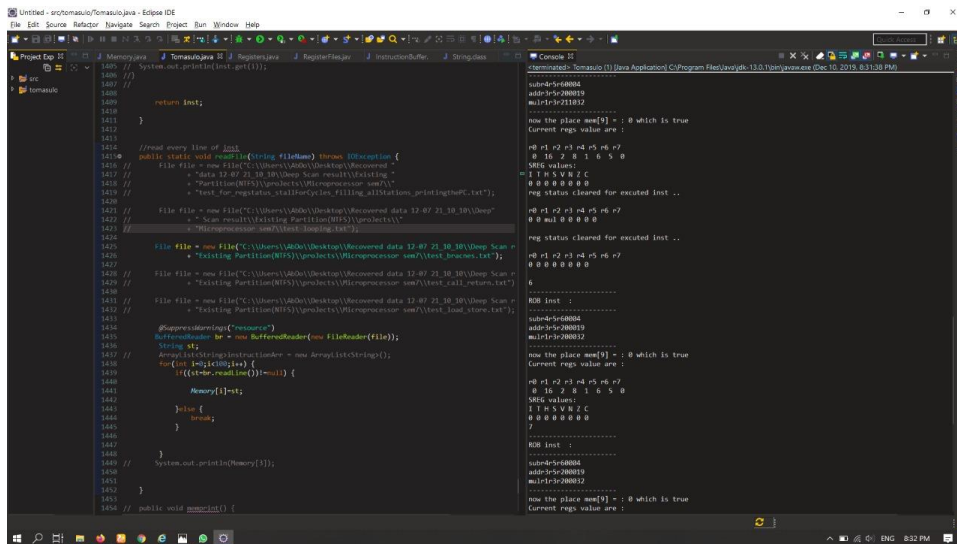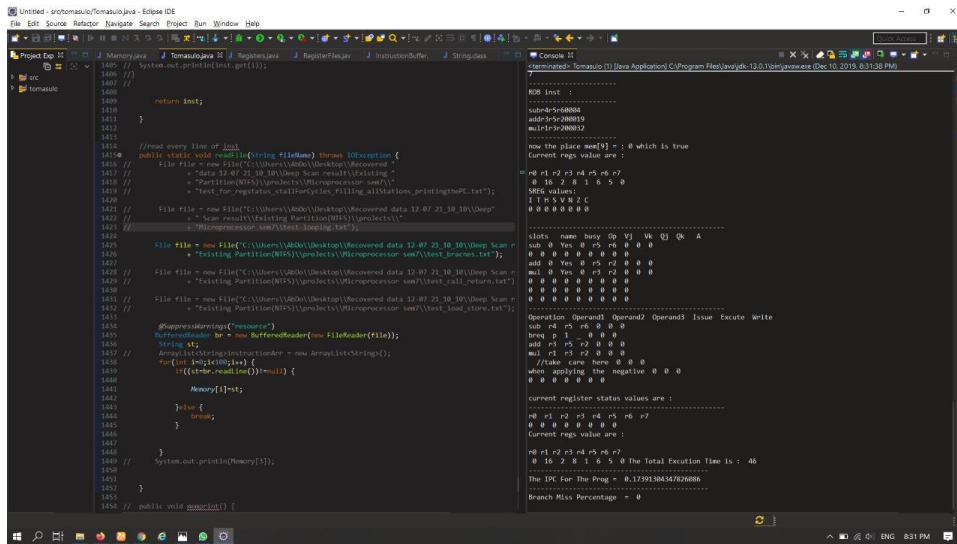
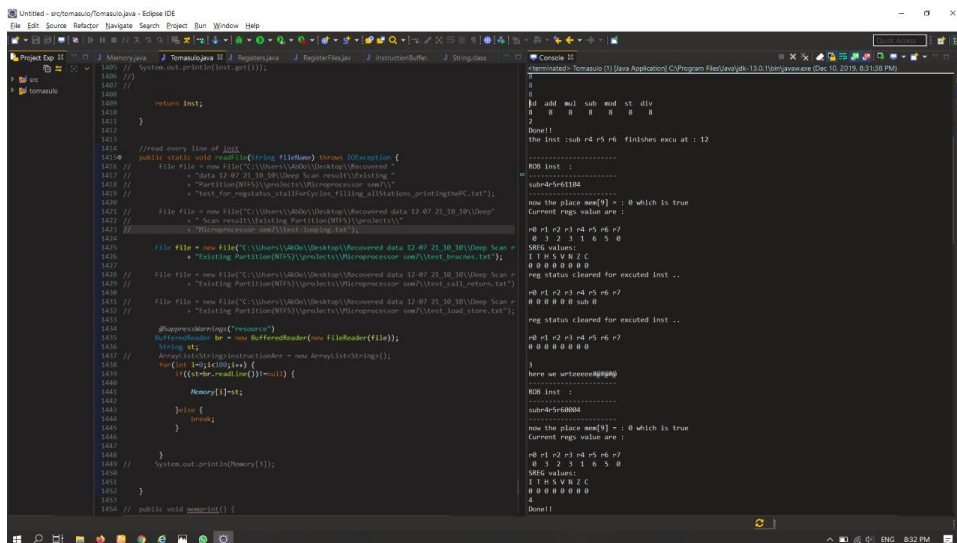   1- test_for_regstatus_stallForCycles_filling_allStations_printingthePC

2-infinte looping: of course it could be modified in the code in the text to but a limitation

3-branching test

```java
// System.out.println(inst.get(i));
//
//

        return inst;

    }

    //read every line of inst
    public static void readFile(String fileName) throws IOException {
//      File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered
//              + "data 12-07 21_10_10\\Deep Scan result\\Existing "
//              + "Partition(NTFS)\\projects\\Microprocessor sem7\\"
//              + "test_for_regstatus_stallForCycles_filling_allStations_printingthePC.txt");

//      File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep
//              + " Scan result\\Existing Partition(NTFS)\\projects\\"
//              + "Microprocessor sem7\\test-looping.txt");

        File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep Scan r
                + "Existing Partition(NTFS)\\projects\\Microprocessor sem7\\test_bracnes.txt");

//      File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep Scan r
//              + "Existing Partition(NTFS)\\projects\\Microprocessor sem7\\test_call_return.txt");

//      File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep Scan r
//              + "Existing Partition(NTFS)\\projects\\Microprocessor sem7\\test_load_store.txt");

        @SuppressWarnings("resource")
        BufferedReader br = new BufferedReader(new FileReader(file));
        String st;
        ArrayList<String>InstructionArr = new ArrayList<String>();
        for(int i=0;i<100;i++) {
            if((st=br.readLine())!=null) {

                Memory[i]=st;

            }else {
                break;
            }

        }
//      System.out.println(Memory[3]);

    }

//  public void memprint() {
```

Console output (screen 1):

```
ROB inst :
-------------------
subr4r5r60004
addr3r5r200019
mulr1r3r200032
-------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  16 2 8 1 6 5 0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0

slots   name  busy  Op  Vj  Vk  Qj  Qk  A
sub  0  Yes  0  r5  r6  0  0  0
0  0  0  0  0  0  0  0
add  0  Yes  0  r5  r2  0  0  0
0  0  0  0  0  0  0  0
mul  0  Yes  0  r3  r2  0  0  0
0  0  0  0  0  0  0  0
0  0  0  0  0  0  0  0

Operation Operand1 Operand2 Operand3 Issue Excute Write
sub  r4  r5  r6  0  0  0
breq  p  1 _ 0  0  0
add  r3  r5  r2  0  0  0
mul  r1  r3  r2  0  0  0
   //take  care  here  0  0  0
when  applying  the  negative  0  0  0

current register status values are :
---------------------------------------------
r0  r1  r2  r3  r4  r5  r6  r7
0  0  0  0  0  0  0  0
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  16 2 8 1 6 5 0 The Total Excution Time is :  46
---------------------------------------------
The IPC For The Prog =  0.17391304347826086
---------------------------------------------
Branch Miss Percentage  = 0
```

Console output (screen 2):

```
subr4r5r60004
addr3r5r200019
mulr1r3r211032
-------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  16 2 8 1 6 5 0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 mul 0 0 0 0 0
reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 0 0 0 0 0 0

6
-------------------
ROB inst :
-------------------
subr4r5r60004
addr3r5r200019
mulr1r3r200032
-------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  16 2 8 1 6 5 0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
-------------------
ROB inst :
-------------------
subr4r5r60004
addr3r5r200019
mulr1r3r200032
-------------------
now the place mem[9] = : 0 which is true
Current regs value are :
```

```java
	System.out.println(inst.get(i));
//
//

		return inst;
	}

	//read every line of inst
	public static void readFile(String fileName) throws IOException {
//		File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered "
//				+ "data 12-07 21_10_10\\Deep Scan result\\Existing "
//				+ "Partition(NTFS)\\projects\\Microprocessor sem7\\"
//				+ "test_for_regstatus_stallForCycles_filling_allStations_printingthePC.txt");

//		File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep"
//				+ " Scan result\\Existing Partition(NTFS)\\\\projects\\\\"
//				+ "Microprocessor sem7\\test_looping.txt");

		File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep Scan r"
				+ "Existing Partition(NTFS)\\projects\\Microprocessor sem7\\test_bracnes.txt");

//		File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep Scan r"
//				+ "Existing Partition(NTFS)\\projects\\Microprocessor sem7\\test_call_return.txt")

//		File file = new File("C:\\Users\\AbDo\\Desktop\\Recovered data 12-07 21_10_10\\Deep Scan r"
//				+ "Existing Partition(NTFS)\\projects\\Microprocessor sem7\\test_load_store.txt");

		@SuppressWarnings("resource")
		BufferedReader br = new BufferedReader(new FileReader(file));
		String st;
		ArrayList<String> instructionArr = new ArrayList<String>();
		for(int i=0;i<100;i++) {
			if((st=br.readLine())!=null) {

				Memory[i]=st;

			}else {
				break;
			}


		}
		System.out.println(Memory[i]);

	}

//	public void memprint() {
```

Console output (screenshot 1):
```
sub r4 r5 r6 0004
addr3r5r211019
--------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  3  2  0  1  6  5  0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 add 0 0 0 0 0

reg status cleared for excuted inst

r0 r1 r2 r3 r4 r5 r6 r7
0 0 0 0 0 0 0 0

5
Done!!
the inst :mul r1 r3 r2  finishes excu at : 40

ROB inst :
--------------------
sub r4 r5 r6 0004
addr3r5r200019
mul r1r3r211032
---------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0 16  2  8  1  6  5  0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 mul 0 0 0 0 0

reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 0 0 0 0 0 0
```

Console output (screenshot 2):
```
0 0 0 0 0 0 0 0

3
here we wrteeeee#@#@#@

ROB inst :
--------------------
sub r4 r5 r6 0004
---------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  3  2  3  1  6  5  0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
4
Done!!
the inst :add r3 r5 r2  finishes excu at : 27

ROB inst :
--------------------
sub r4 r5 r6 0004
addr3r5r211019
---------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  3  2  8  1  6  5  0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 add 0 0 0 0 0

reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 0 0 0 0 0 0

5
Done!!
the inst :mul r1 r3 r2  finishes excu at : 40
```

Console output (screenshot 3):
```
ld add mul sub mod st div
0  0  0  0  0  0  0
2
Done!!
the inst :sub r4 r5 r6  finishes excu at : 12

ROB inst :
--------------------
sub r4 r5 r6 1104
---------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  3  2  3  1  6  5  0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 0 0 0 0 sub 0

reg status cleared for excuted inst ..

r0 r1 r2 r3 r4 r5 r6 r7
0 0 0 0 0 0 0 0

3
here we wrteeeee#@#@#@
ROB inst :
--------------------
sub r4 r5 r6 0004
---------------------
now the place mem[9] = : 0 which is true
Current regs value are :

r0 r1 r2 r3 r4 r5 r6 r7
 0  3  2  3  1  6  5  0
SREG values:
I T H S V N Z C
0 0 0 0 0 0 0 0
4
Done!!
```

4-call and return

5- load and store

5-we did :

(brlo c1,brsh c=0,breq z=1,brmi n=1,add,com,mod,div,ldi,sub,load,store,call ,return,looping ,exit)

😊

...