From last time

Explain the differences between LDR instructions & LDM instructions. (4 marks)

LDR loads 1 value from memory into a register. LDM can load several values into registers from adjacent memory locations.

Explain the effect of the following ARM instruction on the registers involved and on memory: (2 marks)

\[ \text{STMFD SP!, \{R0, R3-R5\}} \]

SP is decreased by 16 (4 words of 4 bytes). Copies of registers R0, R3, R4, R5 are stored in memory locations [SP], [SP+4], [SP+8], [SP+12]
A method is passed two integer arguments via the stack, which it adds together, and returns the result in R0. The method also puts the Link Register on the stack (e.g. because it also calls “println” – but don’t include the code for this call in your answer). Give the ARM code required for a simple implementation of the method and draw a diagram of the resulting stack frame. (4 marks)

```
PUSH {R1,LR}
LDR  R0, [SP,#8]
LDR  R1, [SP,#12]
ADD  R0, R0, R1
LDR  R1, [SP],#4
LDR  PC, [SP],#12
```

Diagram:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>← SP</td>
<td></td>
</tr>
<tr>
<td>LR</td>
<td></td>
<td>SP+4</td>
</tr>
<tr>
<td>param1</td>
<td></td>
<td>SP+8</td>
</tr>
<tr>
<td>param2</td>
<td></td>
<td>SP+12</td>
</tr>
</tbody>
</table>
Question: ARM code for if-else version?

Note: System.out.print("2"); becomes: MOV R0, #'2' then SVC 0

LDR R0, number

tryfour

CMP R0, #0
BNE trytwo
MOV R0, #'0'
SVC 0
B end

trytwo

CMP R0, #2
BNE tryfour
MOV R0, #'2'
SVC 0
B end

default

MOV R0, #'?'
SVC 0
B end

end
Question

What is the ARM code for the start of this switch statement:

```java
switch (number) {
    case –1:
        System.out.print("–1");
        break;
    case 0:
        System.out.print("0");
        break;
    case 1:
        System.out.print("1");
        break;
    default:
        System.out.print("?");
        //break;
}
```
Question – is there a better answer?

(repeat previous question, but try adding 1 to number)

switch (number+1) {
    case 0:
        System.out.print("–1");
        break;
    case 1:
        System.out.print("0");
        break;
    case 2:
        System.out.print("1");
        break;
    default:
        System.out.print("?");
        //break;
}

LDR R0, number
ADD R0, R0, #1
CMP R0, #2
BHI default

ADR R1, table
LDR PC, [R1,R0,LSL #2]

DEFW print_1
DEFW print0
DEFW print1