Problem 1. A syntax chart is given as follows:

```
S               a               S                    T
     c
T               b              S                     T
     c
```

(1) Write a recursive descent parser for this chart in C, Pascal, or Java. A pseudo code notation is allowed, similar to the notes. 6 marks

(2) The trace for the string acc! where ! is an end marker is given as follows:

<table>
<thead>
<tr>
<th>History</th>
<th>Input string</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>acc!</td>
<td>Consume a</td>
</tr>
<tr>
<td>S</td>
<td>cc!</td>
<td>Enter S</td>
</tr>
<tr>
<td>SS</td>
<td>cc!</td>
<td>Consume c</td>
</tr>
<tr>
<td>SS</td>
<td>c!</td>
<td>Exit from S</td>
</tr>
<tr>
<td>S</td>
<td>c!</td>
<td>Enter T</td>
</tr>
<tr>
<td>ST</td>
<td>c!</td>
<td>Consume c</td>
</tr>
<tr>
<td>ST</td>
<td>!</td>
<td>Exit from T</td>
</tr>
<tr>
<td>S</td>
<td>!</td>
<td>Exit from S</td>
</tr>
<tr>
<td>empty</td>
<td>!</td>
<td>Accept</td>
</tr>
</tbody>
</table>

Following this example, trace your parser with the string aaccbcc! 6 marks

Problem 2. The following PL/0 program multiplies a and b and stores the result in z.

```plaintext
const a=4, b=3;
var x,y,z;
procedure P;
begin
  if y>0 then begin
    y:=y-1;  z:=z+x;
    call P;
  end;
end;
begin
  x:=a;  y:=b;
  call P;
  write(z);
end.
```
The corresponding object code is given as follows:

```
0  jmp 0 17
1  jmp 0 2
2  inc 0 3
3  lod 1 4
4  lit 0 0
5  opr 0 12
6  jpc 0 16
7  lod 1 4
8  lit 0 1
9  opr 0 3
10 sto 1 4
11 lod 1 5
12 lod 1 3
13 opr 0 2
14 sto 1 5
15 cal 1 2
16 opr 0 0
17 inc 0 6
18 lit 0 4
19 sto 0 3
20 lit 0 3
21 sto 0 4
22 cal 0 2
23 lod 0 5
24 wrt 0 0
25 opr 0 0
```

(1) Copy this code onto the answer sheet and give a comment to each instruction describing the meaning in the source program. 8 marks

(2) After procedure P is entered twice and the machine instruction at address 12 is executed the stack becomes as follows: 7 marks

```
0 0 0 4 1 4 1 1 23 1 7 16 4 4
```

Explain the meaning of each element.

(3) Show the contents of the stack after you execute instruction 12 for the third time. 7 marks