EEEE-521-621-Lab5

1) From mycourses download on your desktop this pdf file.
2) The goal of this lab is to continue and complete the development of the assembler for your instruction set.
3) In this lab you have to show working **transfer, jump, call, and ret** instructions, with both **forward and backward jump addresses**.
4) 521: Modify the code in **dxp.txt** so that you execute at least one jump unconditional, one conditional if a status bit is 1, and one conditional if a status bit is 0.
5) 621: Same as in 4) above, but in addition use also CALL and RET.
6) If you use the DevC++ development environment, archive your project folder, including any intermediately generated output files.
7) If you are using another compiler/IDE, provide information about these in your report. Include your source code files and meaningful screen shots to proof that your assembler is working.
8) Ultimately, compile a version for release. This should be a standalone executable file. The TA will use this to verify your assembler.
9) If time permits, show your assembler at work to your TA. Note: only the TA will assign a grade.
10) Archive your project.
11) Write your report and upload it along with your archived project(s) in the dropbox on mycourses, as described in the lab policy.
12) In addition, write a concise "User's Guide" for your assembler. Page count is not enforced, but it should contain all necessary and sufficient information for someone to be able to use your assembler – NOT its code!
13) This concludes this week's lab.
14) **Grading**:
    a. 521:
        i. Load and Store = 2 x 3 = 6 points.
        ii. Jump unconditional with forward and backward jump = 2 x 3 = 6 points.
        iii. Jump conditional if status bit is 1 with fwd and bwd jump = 2 x 3 = 6 points.
        iv. Jump conditional if status bit is 0 with fwd and bwd jump = 2 x 3 = 6 points.
        v. Assembler User's Guide = 6 points.
        vi. Total = 30 points.
    b. 621:
        i. Load and Store = 2 x 3 = 6 points.
        ii. Jump unconditional with forward and backward jump = 2 x 3 = 6 points.
        iii. Jump conditional if status bit is 1 with fwd and bwd jump = 2 x 3 = 6 points.
        iv. Jump conditional if status bit is 0 with fwd and bwd jump = 2 x 3 = 6 points.
        v. CALL and RET (fwd call is fine) = 2 x 2 = 4 points.
        vi. Assembler User's Guide = 2 points.
        vii. Total = 30 points.