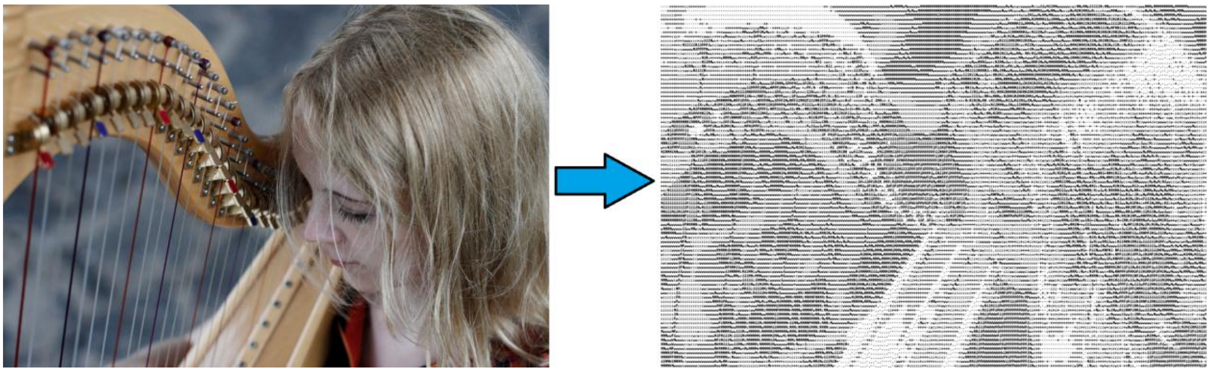
Project Brief: ASCII Art Image Conversion

Your task is to write a computer program in C/C++ that converts source image(s) to ASCII art, i.e. a representation of the original image using ASCII characters only.



The program you create should allow the user to select an image or a sequence of images (e.g. animation frames) and then process them (match pixel regions with ASCII characters of comparable intensity/brightness), saving the results in a format selectable by the user, which should be either as ASCII text files, an image file of ASCII characters (with a selectable text and background colour) or as an image of coloured ASCII characters (using the average colour of the pixels they replace) on a neutral background.

Recommended reading:

- ASCII art. (2016). In Wikipedia, The Free Encyclopedia – http://en.wikipedia.org/wiki/ASCII_art
- Mikolay, M. (2012). A Basic ASCII Art Algorithm – http://mattmik.com/articles/ascii/ascii.html
- Parberry, I. (2011). ASCII Art on a Pixel Shader – http://larc.unt.edu/ian/art/ascii/shader/

Further reading:

- Xu, X., Zhang, L. and Wong, T.-T. (2010). Structure-based ASCII art. In ACM SIGGRAPH 2010 papers – http://dx.doi.org/10.1145/1833349.1778789

Marking Criteria:

1. The quality of the source code and usability of the program. This includes effective use of programming techniques, such as using structured and procedural programming, recursion, relevant control structures and data structures, etc. This will count for 50% of the mark of your assignment.

   The remaining 50% will be determined by:
2. The project report, including Background, Implementation, Results (see submission details below).
3. The visual Impact of the generated artefacts (based on the Results section of your report, and/or submitted generated artefacts, see submission details).
4. Source code documentation (relevant and appropriate comments in the source code).

Submission Details

Your submission needs to include the following:

1. One or more well commented source code files and a *Makefile* that will build the program without errors on any machine in the labs.
2. A PDF user Manual for your application. The User Manual should explain how to initialise and run your program.
3. A PDF report documenting and explaining your application. The report should be approximately 6-8 A4 pages of text and should not exceed 10 A4 pages. The report should contain a section called “Background” or “Introduction” explaining the algorithms, techniques, and ideas used in your project. It should also have a section called “Implementation” explaining the structure of your

program in terms of the implementation of the algorithms and techniques used, describing flow of control, and explain the implementation of the most important functions or procedures. This section should serve to illuminate but not replicate your code. Finally the report should contain a section called “Results” which includes images, or references to images (and video) demonstrating and explaining the results produced by your program.

4. Appropriate sample results (and source data, such as original images where this applies), such as animations or images generated by your program.

Note: this assignment is not a group project assignment but an individual project assignment, and apart from trivial parts of your source code, we expect to see original/different source code and original reports submitted by different students.

Make sure that any 3rd party source code (e.g. found in a book or downloaded from the web) or source code you may have been given by fellow students or members staff is included in comments explaining where the code was found and who is the author – if you received help with parts of your code the person helping you should be acknowledged in a source code comment as well as in your report. Borrowed/3rd party code should be bracketed by comments:

```
/* source from Author Name (http://www.code.ac.uk/web/site.html) starts here: */
int ival(double v) /* return the closest integer */
{
   return ((v - floor(v)) > 0.5) ? (int)ceil(v): (int)floor(v);
}
/* source from Author Name ends here */
```

If the author is unknown state the web-site or location where the source code was found. If the source is modified, but the modifications are small, start and end the comments with “ `source based on “Author Name”` ”.