COMPUTER SCIENCE 110

**INTRODUCTION TO COMPUTER SCIENCE**

**HOMEWORK 1 (Multimedia Python)**

# Fall 2018

**Assigned**: Monday, 9/10 **Due**: **by beginning of class** Monday, 9/24

**Answer the homework questions on a separate page. For each question, indicate the homework part (I or II) and question number. You do not need to turn in this sheet**.

**For the programming portion in part III, follow the directions in that section.**

BD14710_

**Part I (40 points) Introduction, Number Bases** (**show your work for full credit; it’s all right to hand-write the work, but be neat (or type if you can’t be neat): take pride in the product of your efforts!). Your answers must be on a separate sheet of paper; do not turn in your answers on this sheet**.

1. Convert the following **sign-magnitude** numbers from binary to decimal (base 10). Assume that a “1” in the leftmost bit indicates a negative sign. (3 pts.)
   1. 101110000
   2. 000110000011
   3. 10100000
2. Convert the following decimal numbers into binary form using sign-magnitude and 8 bits (including the sign) of representation. (3 pts.)
   1. 244
   2. -42
   3. 9
3. Fill in the following chart with equivalent values in each cell. For example, for the first line, convert the numbers from decimal to binary and then to hexadecimal and octal. **Do not use sign-magnitude notation.** (12 pts.):

|  |  |  |  |
| --- | --- | --- | --- |
| Base 10 | Base 2 | Base 16 | Base 8 |
| **196** |  |  |  |
|  | **1000111** |  |  |
|  |  | **340** |  |
|  |  | 17F | **577** |

1. What is the range of **signed** decimal values that can be represented using each of the following numbers of bits? How many different values are in the range? Express your answers in base 10. (9 points)
   1. One bit
   2. Twelve bits
   3. 4 bytes
2. What do the following ASCII character codes spell? (3 pts.)

01001000 00110100 01110010 01110110 01100101 01110011 01110100 00100000 01000111 01110010 00110011 00110011 01010100 00110001 01101110 01100111 01110011 00100001

1. How would the following string of characters be represented using run-length encoding? What is the compression ratio? (4 pts.)

Reddd, Grrreeeeeeeeeeeen, Bluuuue, Oh my!!!!

1. Huffman codes: Use the Huffman code on page 94 of the CSI textbook to answer the following (6 pts):
   1. Write the Huffman code for the word “FINDERS”
   2. Decipher the Huffman code: 1001110100010000101100101011
   3. Decipher the Huffman code: 100000100111010000101000

BD14710_

**Part II. (25 points) The Last Lecture**

On September 18, 2007, Carnegie Mellon professor and alumnus Randy Pausch, the inventor/driving force/inspiration behind the Alice programming language, delivered a one-of-a-kind last lecture that made the world stop and pay attention. It became an Internet sensation viewed by millions, an international media story, and a best-selling book that has been published in 35 languages. To this day, people everywhere continue to talk about Randy, share his message and put his life lessons into action in their own lives. He passed away in July 2008 of pancreatic cancer.

An excerpt from the talk:

*“Almost all of us have childhood dreams: for example, being an astronaut, or making movies or video games for a living. ...  Sadly, most people don't achieve theirs, and I think that's a shame. I had several specific childhood dreams, and I've actually achieved most of them. More importantly, I have found ways, …, of helping many young people actually \*achieve\* their childhood dreams. This talk will discuss how I achieved my childhood dreams … and will contain realistic advice on how \*you\* can live your life so that you can make your childhood dreams come true, too. “*

So, what are the guidelines for this question?

1. To access the video, go to <http://www.cmu.edu/randyslecture/> and watch the video.
2. Turn in **at least** 1 1/2 pages (typed, 12 pt., double spaced, times new roman or arial font, no larger than 1” margins) discussion of what parts of the video stood out the most.
3. Discuss somewhere in the response the advice his advisor gave to him about dealing with other people.  How did his advisor present this piece of advice and what was another way (less tactful) that he could have said the same thing?
4. What had the most emotional impact on you, personally?  How did this speak to you?
5. Who was the real audience for his lecture?

BD14710_

**Part III (35 points) Python Programming**

Your Python code must follow these guidelines:

* *Always put your name, date, program name, and a general comment describing what your program does at the top of your code. This should be a general statement about how your program behaves.*
* *Variable names should help describe the purpose of a variable.*
* *Function names should help describe the purpose of a function*
* *Every function should be preceded by a comment describing the function’s purpose, input, and output (if any). This is true whether you wrote the function yourself or not.*

Be aware that if you don’t follow these guidelines, you will lose points, **even if your program runs perfectly**.

Write a program named **createSequence()** to create a sequence of the same image at least four times (display the original and at least three modifications). **Your program must be saved in a file called “hw1.py” in your csis110/hw/hw1 folder** You can use any image you want, something you find on the web or a picture of your dog, or whatever – just make sure it’s not the same picture anyone else is using, to your knowledge. The first image you display must be the original, unaltered image. The others must be modified forms of the original image. You can use any of the manipulations we used in lab or that are described in Chapter 4 of the Python book, you can shift or alter colors, make it darker/lighter, or virtually anything else you want. Be creative and have fun with it!! Your program must also pause after each version of the image is displayed, to give the user time to appreciate your work. Finally your program must print the width and height of the image (in pixels) and the number of pixels in the image.

We have given you a working program that does part of this so that you have a starting point: it is printed below. We are not giving you this program online, only on paper, so you will need to type it in.

**NOTE**: A perfectly working program as described above will earn 30 points. To earn the remaining 5 points, you must use your own creativity to go above and beyond those minimal requirements. Examples include additional manipulations of the original picture, original or unique or difficult manipulations, etc. Go nuts!

**Tips**:

1. Type in the working program that we give you below. Get that to work in your account, then change it to use a different picture, add more manipulations, etc.
2. Build on what you’ve learned! There’s actually not much that’s new here, mostly putting together things you’ve already done. If you don’t have your book beside you and your code from the labs in front of you when you’re doing this, then you are going to waste a lot of time.

**TURNING IT IN:**

1. Turn in a printout of your code (the hw1.py file) with the rest of your homework.
2. Put a copy of your hw1.py file and your original image in your csis110/hw/hw1 folder. THIS IS REQUIRED IN ORDER FOR ME TO GRADE IT AND FOR YOU TO EARN FULL CREDIT.

# CSIS110, HW1, Fall 2018, sample code and template

#

# We're giving you this to help you get started. Type this in and

# get it to run, THEN change the picture to one of your choosing

# and complete the rest of the program as required. Make sure your

# program fulfills all the assignment requirements, and have fun!!

# Program creates a sequence of 2 copies of the "bigben.jpg" picture,

# the original plus one with less red.

#

# To run this program: put the picture you manipulate into your

# hw/hw1 folder, load it into JES, run "setMediaPath()" from command

# area to set the media path to your hw/hw1 folder, then

# type "createSequence()".

import time

# Reduce red in picture by 50% (code from book, Program 34)

def decreaseRed(picture):

for pix in getPixels(picture):

value = getRed(pix)

setRed (pix, value \* 0.5)

# Creates a sequence of 2 versions of an image, the original

# then one with less red. Also reports the width of the image.

def createSequence():

# Find original picture (you will use a different picture)

myPic = makePicture (getMediaPath( "bigben.jpg"))

# Display the picture and pause 3 seconds

repaint (myPic)

time.sleep(3)

# Perform the first manipulation of the picture and display it

decreaseRed (myPic)

repaint (myPic)

# other manipulations and requirements go here, after you get the

# initial program working ....

# wrap up: print image statistics

width = getWidth (myPic)

print "this image is " + str(width) + " pixels wide"