# UtPod
# EE 312

### Program Due: Tuesday 4/2/19 at 10:00pm

### 100 points

---

For this program we will be implementing the storage portion of the UtPod (the much smaller version of the iPod). The UtPod will be storing the following information:

- **The head of the linked list (the number of songs is limited only by the amount of memory in the UtPod).**
- **The number of MB of memory will be set when the UtPod is "constructed"**
  - **You must provide a constructor that will set the size of the memory. The maximum is 512MB.**
  - **The user of the class will pass in a size. If the size is greater than 512MB or less than or equal to 0, set the size to 512MB.**
- **For each song we will store the title (string), artist (string), and size (int) in MB**
  - **the title and artist cannot be blank**
  - **the size must be greater than zero**
- **You will need to be able to add and remove songs, show the list of songs, shuffle the list and sort the list.**
- **You must create a destructor that frees the memory for the list of songs.**

**You will create your own header file for the Song class. Here is the header for the UtPod.**

**You will create your own driver program (examples will the shown in class). This one should work with your code to get started.**

**The following code should work as a makefile.**

---

**Things you should do:**

- **Create classes for UtPod and Song with appropriate header files.**
- **Implement public methods in the UtPod class to:**
  - **addSong(Song s)**
  - **removeSong(Song s)**
  - **shuffle()**
  - **showSongList()**
  - **sortSongList()**
  - **getTotalMemory()**
  - **getRemainingMemory()**
- **Implement public methods in the Song class to:**
  - **set and get all instance variables**
  - **overload ==, <,and > operators (used in sorting)**
    - **for < and > use artist, then title, then size**

---

**NOTES:**

- You may have one partner on this project. Your partner must be in the same recitation section.You must use GitHub to store your versions of source code (whether you have a partner or not).
- Your code must compile and run on kamek.ece.utexas.edu.
- The program must be modular and object-oriented, with significant work done by functions. Each function should perform a single, well-defined task. When possible, create re-usable functions. Do not write trivial functions such as a function to read a single value from an input file.
- Make sure you comment the header files well.
- Make sure you write a good driver to test the functionality completely. **You will be graded on how well you tested the classes during the checkout.**

**Turn in:** One zipped file that includes: UtPod.cpp, UtPod.h, Song.cpp, Song.h, UtPodDriver.cpp and a makefile

**Upload**: Turn in a zipped file named prog05_xxxxx.zip where xxxxxx is your UT EID to Canvas.

Be sure to follow the style standards for the course.

rlp 3/11/19