Jun Zhen
CSS 436 Cloud Computing
Program 3

Purpose: This application will allow users to recursively traverses the files of a directory and make a backup onto Amazon Web Service's S3. It also allow users to restore any files uploaded to their S3 account back onto their local machine.

Design: The program starts by taking in an argument. This argument is the directory name that will indicate which directory the user would want to backup. This program takes its argument from the command line. For example, "$ ass3Backup.py testingObjects".

The program will then create a bucket with my student id as a unique identifier. Then it will search through the user's directory and find all of the files. With each file found, the program will check if the object already exists in S3 yet. If object already exist, it will compare the checksum of the current file we want to upload with the object that is already existed in S3. The purpose of this check is to minimize unnecessary operations if object has not been modified. After all the checks, the program will utilizes boto3's function "upload_file()" to upload the file into S3.

The second part of this program is to restore backed up files. The user will also be passing in the directory name as an argument. The program will take the directory name and find it in S3 and utilizing boto3's download_files() function to download all the files onto their local computer.

ass3Backup.py - store directory and its files up to Amazon Web Service's S3 storage

- def makeBucket()
    - Creates a bucket in the aws s3.
    - Return True/False
- def printAllBuckets()
    - Go through s3 and prints out all the buckets and the objects
    - Return void
- def readFile()
    - Helper function that return file.read()
    - Return read() function
- getLocalChecksum()
    - Using hashlib module to get the checksum number of a file and return the value.
    - Return integer
- getS3Checksum()
    - Objects backed up with this program has a checksum value as their metadata. This function will return the checksum value of a s3 object
    - Return integer
- Backup()
    - Using os.walk() to recursively traverse the given folder and uploads all the files into s3.

  - Return void
- checkObjExists()
  - check if objects exists in the bucket. If it exists, then don’t upload otherwise upload files.
  - Return True/False

ass3Restore.py - given a specific directory object, restore all the files belong to that directory object onto a local machine.

- Restore()
  - Utilize os.path() to recursively create the same file structure as the one pulled from S3 for each objects
  - Return void
- checkObjExists()
  - check if objects exists in the bucket. Only pull objects if it exists.
  - Return True/False