

# 327 Proj 6 outline

①

MEN Tester

my Global Cache

// global

start a bunch of threads that  
try to add strings to cache

Thread1 - - - - ThreadN

\ | | /

my global cache

every access synchronized by mutex {  
add  
load, save (uses Data Store \*)  
clean cleans my strings  
getcount

uses Data Store - File to load & save.

Data Store

└ encrypt()  
└ decrypt()

Data Store File

└ load()  
└ save()

Strong\_Database mgc;

encryption  
save w/ encrypt

Crypto\_AES mc ("The Rollo's");

DataStore\_File mf("temp.txt", &mc);  
mgc.save(&mf);

save w/o encrypt

DataStore\_File mf("temp.txt", 0);  
mgc.save(&mf)

if 0  
then no  
encrypt  
done.

Data Store

does all encrypt/decrypt

bool

```
... :: encrypt(string &myString) {
```

```
    if (myCrypto) {
```

```
        return (myCrypto -> encrypt(myString));
```

```
    return false
```



# DataStore\_File

CPSC 327  
Project 6

you can pass a crypto object to DataStore\_File or not. Use initializer list to set it if base class constructor. If there encrypts if not does not

DataStore (ABC load, save, PVF)

pure virtual function

## DataStore\_File

save()

load()

for item in vector

for item in file

temp = item → serialize();

decrypt (item)

encrypt (temp)

// use string Data;

// write out item to file

// parse Data to

// get val, use count

// put on vector

initializer list

DataStore\_File ( ..... ): DataStore (crypto) ----