**Spider Pseudo-code – Arrest Records**

1) Application start with args

* **State(s):** One or more states to gather records for. “All” will retrieve records for all enabled states.

*Required*

* **Filter:** If provided, program will create a separate sheet with only filtered records. None, alcohol or traffic.

*Optional: defaults to none*

* **Max Number of Results:** Stop crawling (per state) after this many records are retrieved.

*Optional: defaults to 99,999*

* **Start from last crawled:** Only crawls until it reaches the last retrieved record from previous run. If false, will check for saved uncrawled records from previous run and retrieve only them. If none exist, will crawl as if for the first time.

*Optional: defaults to True*

2) The appropriate state-specific engines (logic to parse a particular site) are engaged depending on state(s) chosen. A thread will be started for each and will run concurrently.

3) Engines

3a) Arrests.org engine

1. Previously crawled/uncrawled records are read into memory.
2. Records in files from previous run are backed up and a new workbook is created
3. Site scraping begins
   1. Connect to most recent results page for state
      1. On failure, retry with a new user-agent up to max number of attempts. Else bail
   2. If **Start from last crawled** = false and previously uncrawled records exist (indicates a failure during previous run that prevented all records from being crawled and saved)
      1. Build a specific list of record pages to crawl and proceed to Step iv)
   3. Else
      1. Determine urls of results pages to crawl based on **last crawled record** (if not null) and retrieve the html docs in a random order, also gathering misc pages to prevent obvious pattern
      2. Scrape all record detail urls from results pages as well as various misc pages
4. Record scraping begins
   1. Retrieve the crawled records from memory to start list
   2. Randomize list of record urls
   3. For each url, until **max number of results** is reached or list is exhausted
      1. Retrieve the html doc
         1. On failure, retry with a new user-agent up to max number of attempts. Else proceed to Step ??
      2. If a misc page
         1. Ignore and sleep for half-interval
      3. Else
         1. Scrape record for information
            1. If state on record doesn’t match state being crawled

Skip (Arrests.org doesn’t always sync counties with states e.g. Polk County – WI, IA, MN, etc)

* + - * 1. Else

Add record to list in memory

Save record to spreadsheet

Create a copy of workbook

Add record in next empty row

Delete original file

Rename copy to original name

Mark as CRAWLED in memory

* + - 1. If exception occurs, sleep for half-interval, otherwise sleep for full interval
  1. Finalize the output
     1. Sort list of crawled records by county
     2. If **filter** != None
        1. Parse through list to get a list of filtered records
        2. Split into lists by county
        3. Create spreadsheet with tabs for each county
           1. If spreadsheet already exists, backup and save as new file
     3. Split unfiltered list into lists by county
     4. Backup spreadsheet and save new spreadsheet with tabs for each county

1. Send e-mail with results

3b) DesMoinesRegister.org

1. Previously crawled/uncrawled records are read into memory.
2. Records in files from previous run are backed up and a new workbook is created
3. Site scraping begins
   1. For each county
      1. Connect to web service for json response, limiting result count to **max number of results**/number of counties
         1. On failure, retry with a new user-agent up to max number of attempts. Else bail
      2. Parse response for list of record detail urls
      3. Record scraping begins
      4. Retrieve the crawled records from memory to start list
         1. For each url
            1. Retrieve html doc
            2. Scrape record for information
            3. Add to record list in memory
            4. Save record to spreadsheet

Create a copy of workbook

Add record in next empty row

Delete original file

Rename copy to original name

* + - * 1. Sleep for full interval

1. Finalize the output
   1. Sort list of crawled records by county
   2. If **filter** != None
      1. Parse through list to get a list of filtered records
      2. Split into lists by county
      3. Create spreadsheet with tabs for each county
         1. If spreadsheet already exists, backup and save as new file
   3. Split unfiltered list into lists by county
   4. Backup spreadsheet and save new spreadsheet with tabs for each county
2. Send e-mail with results