

OS FINAL PROJECT

Made by Ahmed Kasteer and Waleed Ashraf



20F-0336, 20F-0173

Course Instructor: Sir HASEEB ARSHAD

Teacher Assistant: Ayyan Shahid

Code for the Project Web Crawler.

C++

#include <iostream>

#include <curl/curl.h>

#include <fstream>

#include<string>

#include<pthread.h>

using namespace std;

string s;

void code(const char\* url);

void fetch\_https1(string file);

void \*threadfunction(void \*arg) {

cout<<s<<endl;

const char\* ccx = s.c\_str();

code(ccx);

fetch\_https1("code.txt");

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

return 0;

}

void fetchhttpsbymutithreading(string file)

{

//thread is created

ifstream in(file);

int numberofthreads = 0;

//string s;

while (!in.eof())

{

getline(in, s);

if (in.eof())

break;

numberofthreads++;

}

in.close();

pthread\_t a\_thread[numberofthreads]; //thread declaration

ifstream in2(file);

for (int i = 0; i < numberofthreads; ++i)

{

getline(in2, s);

if (in2.eof())

break;

pthread\_create(&a\_thread[i], NULL,threadfunction, NULL);

}

for (int i=0;i<numberofthreads;i++)

{

pthread\_join(a\_thread[i], NULL);

}

in2.close();

}

void fetch\_https(string file)

{

bool check = true;

if (file != "code.txt")

{

ofstream out("websitehttps.txt");

out.close();

check = false;

}

cout << "New Site Crawling....." << endl;

ifstream in(file);

char ch;

string https = "";

while (!in.eof())

{

in >> ch;

if (!in.eof())

{

https = "";

if (ch == '"')

{

in >> ch;

if (ch == 'h')

{

https += ch;

in >> ch;

if (ch == 't')

{

https += ch;

in >> ch;

if (ch == 't')

{

https += ch;

in >> ch;

if (ch == 'p')

{

https += ch;

in >> ch;

if (ch == 's')

{

https += ch;

while (ch != '"' && !in.eof())

{

in >> ch;

if (ch == '"')

break;

https += ch;

}

if (check==true)

{

//cout << https << endl;

ofstream out("websitehttps.txt",ios::app);

out << https << endl;

}

else

cout << "nothing" << endl;

}

}

}

}

}

}

}

}

}

void fetch\_https1(string file)

{

bool check = true;

if (file != "code.txt")

{

ofstream out("downloaded.txt");

out.close();

check = false;

}

cout << "New Site Crawling....." << endl;

ifstream in(file);

char ch;

string https = "";

while (!in.eof())

{

in >> ch;

if (!in.eof())

{

https = "";

if (ch == '"')

{

in >> ch;

if (ch == 'h')

{

https += ch;

in >> ch;

if (ch == 't')

{

https += ch;

in >> ch;

if (ch == 't')

{

https += ch;

in >> ch;

if (ch == 'p')

{

https += ch;

in >> ch;

if (ch == 's')

{

https += ch;

while (ch != '"' && !in.eof())

{

in >> ch;

if (ch == '"')

break;

https += ch;

}

if (check==true)

{

//cout << https << endl;

ofstream out("downloaded.txt",ios::app);

out << https << endl;

}

else

cout << "nothing" << endl;

}

}

}

}

}

}

}

}

}

void code(const char\* url)

{

CURL \*easyhandle = curl\_easy\_init();

curl\_easy\_setopt(easyhandle, CURLOPT\_URL, url);

FILE \*file = fopen("code.txt", "w");

curl\_easy\_setopt(easyhandle, CURLOPT\_WRITEDATA, file);

curl\_easy\_perform(easyhandle);

curl\_easy\_cleanup(easyhandle);

}

int main()

{

const char\* url="https://www.google.com/";

code(url);

fetch\_https("code.txt");

fetchhttpsbymutithreading("websitehttps.txt");

return 0;

}

Screenshots

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated