NOVA SCHOOL OF SCIENCE & TECHNOLOGY

José Costa <jppi.costa@campus.fct.unl.pt>

Redes de Computadores 23/24 TP4: Video streaming and HTTP partial requests

Google Forms <forms-receipts-noreply@google.com> 28 de novembro de 2023 às 00:06
Para: jppi.costa@campus.fct.unl.pt

Obrigado por preencher o formulário Redes de Computadores 23/24 TP4: Video streaming and HTTP partial requests

Foi recebido o seguinte.

Editar resposta

Redes de Computadores 23/24 TP4: Video streaming and HTTP partial requests

This is the **form for students to deliver** the **TP4 - 4th Frequency Work Assignment "*Video Streaming and HTTP Partial Requests* " (**for frequency evaluation). In this work the goal is to implement a solution (based in a multithread proxy component) to obtain by HTTP partial requests (or ranges) of segments of movie tracks (given movies with different resolutions provided by a given HTTP content server) and to deliver the contents to a player component (composed by a python player interfacing with media player applications hat can play the movie from the streamed segments, following the requirements of the TP4 statement (Enunciado do Trabalho TP4).
Deadline for the work delivery: 10:30 on 28th November, 2023.
In the deadline date/time your submission is only considered complete if you submit :
- **This form**, filling the informations and answers to questions related to your implementation of TP3
- **The implementation code (in a ZIP archive containing the implementation of client and server), submitted via the system moodle.fct.unl.pt**

Important Note: The ZIP archive (submitted via moodle.fct.unl.pt, tag/course RC2324, See TPC4) must be submitted in the following way:
- Name of the archive: **RC-TPC4-NNNNN.zip** (for individual submissions, NNNNN the Student Number)
- Name of the archive: **RC-TPC4-NNNNN-MMMMM.zip** (for group submissions), NNNNN and MMMMM the student numbers of a group.

O seu email (**jppi.costa@campus.fct.unl.pt**) foi registado quando enviou este formulário.

The TP3 work delivery corresponds to: *

- ( ) Individual work (developed by 1 student)
- (•) Group work (developed by 2 students)

I/We declare that the answers to the questions in this form relate with my/our own implementation. Moreover, I/We declare the authorship of the code we will submit for evaluation, according to the academic ethical rules and non-plagiarism of submitted work as expressed in the ethical conde: UNL: https://www.unl.pt/ensino/direitos-e-deveres-dos-estudantes , DR — N.º 245 — 19 de dezembro de 2014, Despacho n.º 15464/2014, and particularly in Art.10º, points 2 to 6. *

- (•) YES

Student identification

Fill the name and student number below (either individual work or the 1st member of the group in the case of group work)

Name *

Catarina Gonçalves Costa

Student Number *

62497

Student registered in the following practical class *

- ( ) Turno 1 (Practical class P1, Thu 9h-11h)

- ○ Turno P2 (Practical class P2, Thu 11h-13h)
- ◉ Turno P3 (Practical class P3, Thu 14h-16h)
- ○ Turno P4 (Practical class P4, Thu 16h-18h)
- ○ Turno P5 (Practical class P5, Fri 9h-11h)
- ○ Turno P6 (Practical class P6, Fri 11h-13h)

Student identification (2nd element of group). Only fill in if work is carried out in group

Fill the name and student number of member 2 of the group below in the case of group work). In the case of individual work, leave it blank.

Name (2º element of group)

José Pedro Pires Costa

Student Number (2º element of group)

62637

Student registered in the following practical class

- ○ Turno 1 (Practical class P1, Thu 9h-11h)
- ○ Turno P2 (Practical class P2, Thu 11h-13h)
- ◉ Turno P3 (Practical class P3, Thu 14h-16h)
- ○ Turno P4 (Practical class P4, Thu 16h-18h)
- ○ Turno P5 (Practical class P5, Fri 9h-11h)
- ○ Turno P6 (Practical class P6, Fri 11h-13h)

SECTION of QUESTIONS on Generic Objectives Achieved in TP4

In the following answers you will state the characterization of your implementation and achieved goals according to the TP4 requirements (see the TP4 statement for the required functionality). For your answers you must consider your implementation achievements, your tests, your experimental observations, and the state your implemented goals, and how your work addressed the reference of requirements in the TP4 statement (*Enunciado do Trabalho*) and the code submitted for evaluation

Generic goals achieved. Considering the requirements (TP4 statement/Enunciado do Trabalho), my implementation ... (select the case or cases)

	Yes	No	Partially	Unstable
Supports all the requirements	☑	☐	☐	☐
I am convinced and can demonstrate that I have implemented all the work requirements	☑	☐	☐	☐
Works correctly, it is ready for demonstration	☑	☐	☐	☐
It works, with the two movies and all tracks (different resolutions)	☑	☐	☐	☐
In my tests, I play correctly the streamed movies with my media player application	☑	☐	☐	☐
My proxy is multithreaded (using a producer and a consumer thread) as required	☑	☐	☐	☐
My proxy producer thread obtains correctly from the HTTP content server the manifest.txt files for any movie and track resolution	☑	☐	☐	☐
My proxy producer thread obtains by HTTP from the given content server segments of tracks, using ranges in GET requests	☑	☐	☐	☐
My proxy consumer thread can send the obtained segments from the producer thread to the player	☑	☐	☐	☐
The segments sent from the proxy consuer thread are correctly sent to the given player.py pythin program	☑	☐	☐	☐

About the way your proxy is launched

My proxy (program proxy.py in your submitted archive to Moodle) is launched in the following way (exactly as required in the TP4 statement/*Enunciado do trabalho*):

python proxy.py baseURL movieName track

Ex
python proxy.py http://localhost:9999 coco 2
or
python proxy.py http://localhost:9999 dante 4
*

- (●) Yes, exactly as above
- ( ) No, it is launched in a different way

If you selected "***No, it is launched in a different way"*** in the previous question, how do you launch your proxy ? Exemplify how you launch your proxy in the following situations:
a) if you want to launch the proxy for streaming the movie "coco" using the track encoded with the minimum resolution
b) if you want to launch the proxy for streaming the movie "dante", using the track encoded with the maximum resolution

.....

SECTION with QUESTIONS to describe your development and testing environment

In the following questions you characterize your development environment (OS and media player applications used to develop and test your TPC4 implementation)

Select the case of your development and testing environment, selecting the OS (or OSes if used more than one in your tests) *

	Yes	No
Linux OS, all components running in localhostt	☑	☐
Windows OS, all components running in localhost	☑	☐
MacOS, all components running in localhost	☑	☐
None of above	☐	☑

If you selected "None of above" in the last question, clarify your development and testing environments. If this is not the case, left the answer in blank

Select the case of your media player application used in your tests/experimental observation

- [ ] VLC
- [x] MPLAYER
- [ ] MPV
- [ ] Other
- [ ] I didn't test with a media player application

If you selected "Other" in the last question, what media player you used ? If this is not the case, left the answer in blank

If you cannot see the streamed movies playing in your media player application the streamed movies, are you sure that the proxy can obtain all the segments correctly ? Note that to be sure and to demonstrate this, you must verify the the proxy can write in a local file all segments obtained from the content server for a specific track and you must test that you can obtain correctly a file with the same content as you can obtain from the server with your browser

If you didn't complete the requirements, summarize what you did and clarify what is not implemented or what is not working correctly. If this is not the case, leave the answer in blank

SECTION with questions to answer, ONLY IN THE CASE that you cannot see the streamed movie in the media player application (via the player.py) program. If this is not the case, you can go the the following SECTION, leaving te answers of this SECTION in blank

You will describe in he following answers the situation resulting from your debugging, observation tests, evidences, or diagnosing verifications, why your implementation (proxy) is not able to stream and play the streamed movie tracks in the used media player application.

SITUATION 1): ABOUT THE PROXY IMPLEMENTATION, DEBUGGING AND VERIFICATIONS. Are you sure that your proxy (producer thread and consumer thread) is working well, with the proxy producer thread getting correctly the ranges related to the segments from the content server and the consumer thread sending correctly the segments to the player.py program. CAN SELECT MORE THAN ONE OPTION

- [ ] Yes, I think so, even that I cannot see the streamed movie playing in the media player
- [ ] No, the proxy implementation is not correct from my observation tests
- [ ] I'm not sure, and I don't know really if the proxy is doing all the tasks correctly
- [ ] From my tests, evidences and observations, the problem is in the player.py
- [ ] From my tests, evidences and observations, the problem is in the media player
- [ ] I don't know really where the problem is because I don't have detailed debugging tests
- [ ] My media player can play the MP4 files (in the server), if I test to play the files directly with the media player
- [ ] My media player cannot play the MP4 files (in the server), if I test to play the files directly with the media player

SITUATION 2) ABOUT THE VERIFICATIONS OF THE PROXY PRODUCER THREAD

Are you sure that your proxy (producer consumer) is doing the task correctly by processing the manifest.file and requesting/obtaining correctly the segments of each movie track ? You can select more than one option, according to your verifications

- [ ] a) I don't know really if the proxy producer thrad is working well

- [ ] b) I didn't verify, I have no evidence that the producer thread works correctly
- [ ] c) Yes, I am sure. If I write all the segments obtained from the producer threat) to a file , I see that this file is exactly equal to the MP4 track in the content server.
- [ ] d) I am sure the proxy producer thread is working well from other debugging tests I conducted

If you selected d) in previous question, describe what evidence or debugging tests you have to prove it

SITUATION 3) ABOUT THE VERIFICATIONS OF THE PROXY CONSUMER THREAD AND player.py :

Are you sure that your player.py is receiving correctly from the proxy consumer thread all the segments of each movie track obtained by the producer thread? You can select more than one option, according to your verifications.

- [ ] a) I don't know if the proxy consumer thread is sending correctly the segments to player.py
- [ ] b) I didn't verify in detail or with any evidence if the consumer thread is working correctly
- [ ] c) Yes, I am sure. In player.py program I write the received segments to a file , and I see that this file is exactly equal to the MP4 track in the content server.
- [ ] d) Yes, I am sure the player receives correctly all the segments, because I have this verification proof

If you selected d) in the previous question, explain what evidence or verification proof you have to be sure that the player.py program receives correctly from the proxy consumer thread all the segments that you need to pass to the mpeg player application ?

Conclusions and final remarks

In this section you must present conclusions and final remarks, from your experience about the implementation of TP4. You can emphasize any features, highlights of your implementation, or any experimental observations from your development and experimental tests.

Write here your comments, conclusions and remarks