

Opis sistema

Na slici 1 ilustrovana je arhitektura sistema za automatsko snimanje audio/video (AV) sadržaja koji se sastoji od:

- AV upravljačkog sistema,
- proizvoljnog broja AV snimača,
- mrežnog podsistema za čuvanje podataka (mrežni disk),
- mrežnog komutatora (eng. Switch), i
- krajnjeg korisnika koji koristi ovaj sistem.

Korisnik AV upravljačkom sistemu šalje zadatke u kojima se definiše sledeće:

1. ime AV snimača, tj. njegovu oznaku/broj,
2. početak snimanja,
3. dužina snimanja,
4. tip snimanja (audio, video ili audio i video),
5. prioritet samog zadatka.



Slika 1 – Arhitektura sistema za automatsko snimanje AV sadržaja

Zadatak

Potrebno je implementirati raspoređivač zadatka čija funkcionalnost se zasniva na odabiru zadataka sa najvećim prioritetom, tj. implementirati prioritetni raspoređivač zadataka (eng. Priority Scheduler).

Tehnička specifikacija

- Pretpostaviti da je broj AV snimača neograničen.
- Zadatak implementirati u posebnom Python modulu *Scheduler.py*.
- Zadatak čija vrednost prioriteta je **najveća** ima **najveći** prioritet.

Obezbediti sledeću aplikativnu spregu:

- `get_next_task(...)` -> `task`
 - Funkcija koja dobavlja zadatak najvećeg prioriteta
 - Sami definišite parametre funkcije
 - Povratna vrednost funkcije je objekat pronađenog zadatka
 - Vremenska performansa funkcije treba da bude kompleksnosti $O(\lg n)$
- `decrease_task_priority (... , new_priority)`
 - Funkcija koja *smanjuje* prioritet jednog od zadataka (veća vrednost prioriteta - > veći prioritet; tj zadatak sa prioritetom 1 ima **najmanji** prioritet)
 - Jedini obavezan parametar je vrednost novog prioriteta, ostale parametre definišite sami
- `add_new_task (... , new_task)`
 - Funkcija koja dodaje novi zadatak u listu zadataka
 - Jedini obavezan parametar je objekat koji opisuje zadatak koji se dodaje
- `get_all_tasks_sorted(...)` -> `list of tasks`
 - Funkcija koja sortira zadatke po prioritetu (od najnižeg ka najvišem)
 - Sami definišite parametre funkcije
 - Povratna vrednost funkcije je lista sortiranih zadataka
 - Vremenska performansa funkcije treba da bude kompleksnosti $O(n \lg n)$

Testirati svaku od implementiranih funkcija u posebnom modulu *main.py*.

Primer skupa testnih podataka

Ulazni podaci:

```
[{priority: 5, start_time: 10, duration: 20, type: audio, recorder: rec1},  
{priority: 3, start_time: 15, duration: 10, type: video, recorder: rec2},  
{priority: 1, start_time: 2, duration: 60, type: audio, recorder: rec3},  
{priority: 6, start_time: 13, duration: 33, type: audio, recorder: rec4},
```

```
{priority: 2, start_time: 8, duration: 48, type: video, recorder: rec5},  
{priority: 4, start_time: 26, duration: 25, type: video, recorder: rec6}]
```

Rezultat raspoređivanja:

```
[{priority: 1, start_time: 2, duration: 60, type: audio, recorder: rec3},  
{priority: 2, start_time: 8, duration: 48, type: video, recorder: rec5},  
{priority: 3, start_time: 15, duration: 10, type: video, recorder: rec2},  
{priority: 4, start_time: 26, duration: 25, type: video, recorder: rec6}  
{priority: 5, start_time: 10, duration: 20, type: audio, recorder: rec1},  
{priority: 6, start_time: 13, duration: 33, type: audio, recorder: rec4}]
```

Rezultat nakon izvršavanja funkcije `get_next_task`:

```
{priority: 6, start_time: 13, duration: 33, type: audio, recorder: rec4}
```

Rezultat nakon izvršavanja funkcije `decrease_task_priority` (prioritet zadatka promenjen sa 1 na 10), a zatim izvršavanja funkcije `get_next_task`:

```
{priority: 10, start_time: 2, duration: 60, type: audio, recorder: rec3},
```

Rezultat nakon izvršavanja funkcije `add_new_task` (dodat je zadatak {priority: 7, start_time: 12, duration: 56, type: audio, recorder: rec7}):

```
[{priority: 10, start_time: 2, duration: 60, type: audio, recorder: rec3},  
{priority: 5, start_time: 10, duration: 20, type: audio, recorder: rec1},  
{priority: 7, start_time: 12, duration: 56, type: audio, recorder: rec7},  
{priority: 3, start_time: 15, duration: 10, type: video, recorder: rec2},  
{priority: 2, start_time: 8, duration: 48, type: video, recorder: rec5},  
{priority: 4, start_time: 26, duration: 25, type: video, recorder: rec6},  
{priority: 6, start_time: 13, duration: 33, type: audio, recorder: rec4}]
```