

Први колоквијум из Скрипт језика

23.11.2017.

Написати Python 3 апликацију која имитира кретање кроз систем датотека. Захтеви апликације:

- (1) Написати класе File и Folder. Сваки фајл има свој назив (стринг), величину (цео број) и родитељски фолдер. Фолдер је фајл који је колекција других фајлова и његова фиксна величина је 4, док се по потреби његова величина израчунава као $4 + \text{величина свих фајлова (укључујући фолдере)}$ у њему. Назив, величина и родитељ примају се у инит методи (за фолдер се не прима величина). Ако се не проследи у инит методи, родитељски фолдер је None, иначе се фајл додаје у колекцију тог родитеља. Ако при иницијализацији неки аргумент није исправан, баца се изузетак са одговарајућом поруком о грешци.
- (1) Надјачати функцију len за фајлове тако да враћа величину фајла. У случају фолдера, треба да враћа величину израчунату по горе наведеној формули. Ова функција се надаље увек користи уместо директног гледања поља velicina код фајлова и фолдера.
- (2) Написати класу Disk, која чува назив (стринг), капацитет (цео број) и свој root фолдер. Назив, капацитет и root примају се у инит методи. Сви типови аргумената морају бити валидни. Тренутна величина root фолдера не сме бити већа од капацитета диска. Ако при иницијализацији неки аргумент није исправан, баца се изузетак са одговарајућом поруком о грешци.
- (2) Написати методу Disk::dodaj_fajl(folder, fajl) која додаје фајл у фолдер. Фајлу се истовремено придружује одговарајући родитељ. Аргументи морају бити одговарајућег типа (Folder, File). Фолдер мора бити већ смештен на диску (root или неки потфолдер) да би позив успео, а фајл не сме бити на диску пре позива. Величина фајла (који може бити и други фолдер) не сме да прекорачи преостали простор на диску. У случају грешке подиже се одговарајући изузетак.
- (1) Надјачати функције str и repr за фајл тако да враћају назив фајла.
- (4) Написати генераторе File::ls(*args) и Folder::ls(*args) који примају произвољне аргументе и генеришу све фајлове из свог подсистема. Аргументи који имају значај кад се проследи су 'R', 'n', 'f', 'v', 'r', при чему 'R' значи рекурзивно, 'f' значи да генерише само фолдере, 'n' значи сортирано по називу фајла растуће, 'v' значи сортирано по величини опадајуће а 'r' значи сортирање у инверзном редоследу ('n' и 'v' су међусобно искључиви, прихвата се онај који је касније наишао у листи аргумената; ако није прослеђен ниједан, 'r' зависи од имплементације, али не изазива грешку). Опис генерисања:
 - Ако се не проследи 'R', генеришу се само фајлови и фолдери на тренутном нивоу
 - Ако се проследи 'R', када се налети на фолдер, треба изгенерисати тај фолдер и онда све фајлове у њему (рекурзивно применити ls са истим аргументима)
 - Ако се проследи 'f', не генеришу се обични фајлови, већ само фолдери; ако се не проследи, генерише се све
 - Ако се проследи неки од аргумената за сортирање, сортира се у одговарајућем редоследу садржај тренутног фолдера, а онда се исто примењује на садржај потфолдера уколико је позив рекурзиван
- (2) Написати методу Disk::ls(putanja, *args), при чему се исписује назив и капацитет диска, диск проналази одговарајућу путању у свом фајл систему и исписује у петљи све што се генерише помоћу генератора ls на тој путањи са прослеђеним аргументима. За сваки фајл у малим заградама поред исписати његову величину, а ако је фолдер исписати и f. Root фолдер је на путањи '/', као у bash-у, а остатак се гради називима фолдера раздвојеним '/'. Видети примере на крају. Ако путања не постоји, подићи изузетак.
- (2) Направити следећи фајл систем на диску 'test' величине 8000 и тестирати помоћу d.ls('/', 'R'), d.ls('/folder1', 'n', 'R', 'r') и d.ls('/', 'f', 'v').

/	d.ls('/', 'R')	d.ls('/folder1', 'n', 'R', 'r')
-folder1		test 8000
-file1 (20)	test 8000	folder1 (4461f)
-folder3	/ (6012f)	folder3 (4f)
-file5 (4000)	folder1 (4461f)	file5 (4000)
-file2 (433)	file1 (20)	file2 (433)
-folder2	folder3 (4f)	file1 (20)
-file3 (70)	file5 (4000)	
-file4 (180)	file2 (433)	
-file6 (881)	folder2 (74f)	
-file7 (412)	file3 (70)	
	file4 (180)	
	file6 (881)	
	file7 (412)	

d.ls('/', 'f', 'v')
test 8000
/ (6012f)
folder1 (4461f)
folder2 (74f)