

SkylineBot

El projecte SkylineBot per GEI-LP (edició primavera 2020).



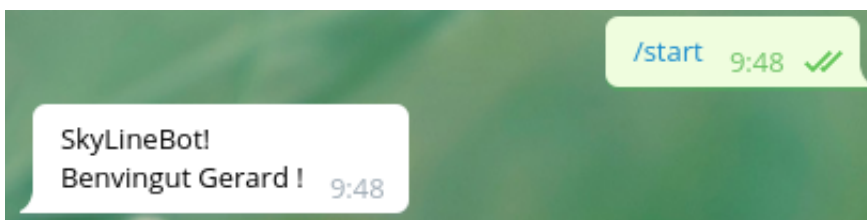
L'*skyline* d'una ciutat mostra una vista horitzontal dels seus edificis.

L'objectiu general de la pràctica consisteix en desenvolupar un chatbot en Telegram per a la manipulació d'*Skylines* via un intèrpret.

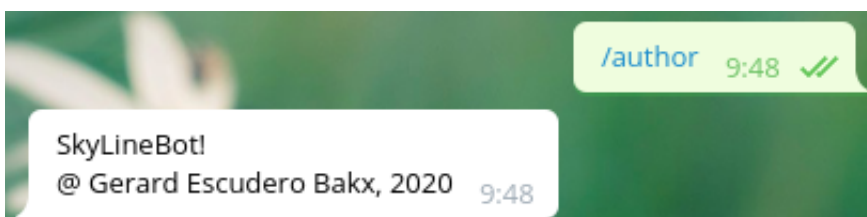
Llenguatge

Abans de començar a detallar el que s'espera, anem a veure un exemple d'interacció amb el nostre chatbot intèrpret.

Exemple d'interacció



La comanda `/start` s'encarrega de fer totes les inicialitzacions pertinents i donar el missatge de benvinguda.

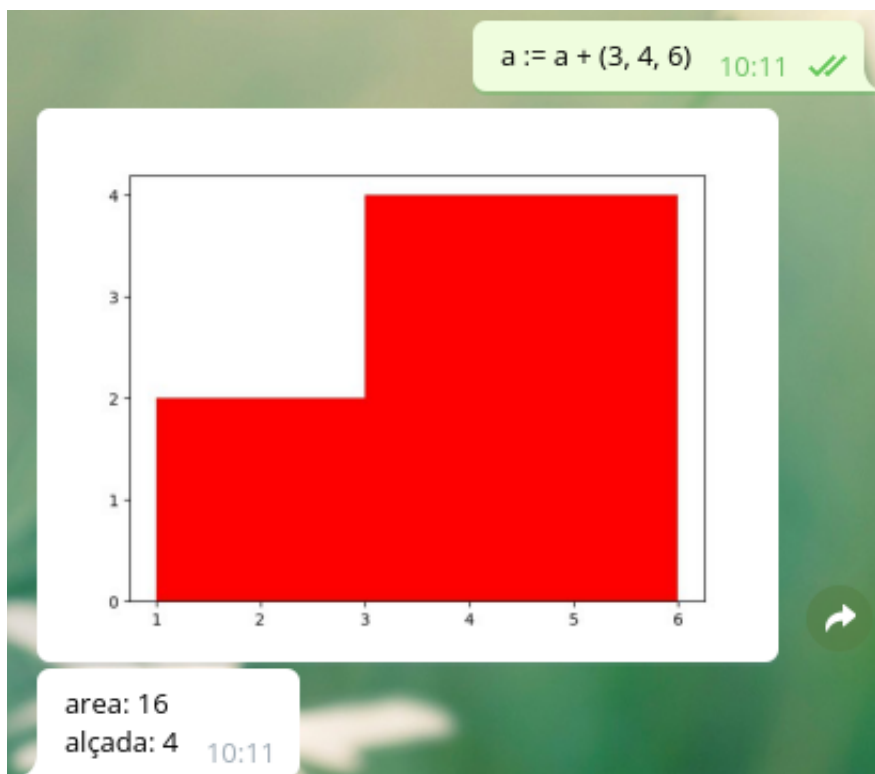


La comanda `/author` ens dona la informació sobre l'autor del bot.



Amb l'expressió `(1, 2, 3)` estem creant un edifici que va desde la posició horitzontal 1 a la 3 i que té una alçada de 2. Amb l'operador `:=` estem assignant l'edifici a l'identificador `a`. De fet, estem creant una skyline amb un únic edifici.

El resultat que dona l'interpret és una representació de l'skyline juntament amb el càlcul de la seva àrea i alçada.



En aquest cas estem afegint un edifici nou (3, 4, 6) a l'skyline anterior mitjançant l'operador $+$.



Amb l'operador $*$ i un nombre natural podem replicar l'edifici tantes vegades com especifiquem.



També podem reflectir un skyline. Noteu que en aquest cas no estem assignant res, només preguntant pel resultat d'una expressió.

Notació del llenguatge

El llenguatge permet els tipus d'operacions següents:

- Creació d'edificis:
 - Simple: $(x_{\min}, \text{alçada}, x_{\max})$ on x_{\min} i x_{\max} especifiquen la posició d'inici i final a la coordenada horitzontal i alçada l'alçada de l'edifici. Ex: $(1, 2, 3)$.
 - Compostos: $[(x_{\min}, \text{alçada}, x_{\max}), \dots]$ permet definir diversos edificis mitjançant una llista d'edificis simples. Ex: $[(1, 2, 3), (3, 4, 6)]$.
 - Aleatoris: $\{n, h, w, x_{\min}, x_{\max}\}$ construeix n edificis, cadascun d'ells amb una alçada aleatòria entre 0 i h , amb una amplada aleatòria entre 1 i w , i una posició d'inici i de final aleatòria entre x_{\min} i x_{\max} .

En la creació d'edificis ha de controlar que x_{\max} sigui més gran que x_{\min} i que no es donin edificis amb alçades negatives.

- Operadors d'skylines:
 - $\text{skyline} + \text{skyline}$: unió
 - $\text{skyline} * \text{skyline}$: intersecció
 - $\text{skyline} * N$: replicació N vegades de l'skyline (vegeu exemple d'interacció).
 - $\text{skyline} + N$: desplaçament a la dreta de l'skyline N posicions.
 - $\text{skyline} - N$: desplaçament a l'esquerra de l'skyline N posicions.
 - $- \text{skyline}$: retorna l'skyline reflectit.

La taula següent mostra la prioritat d'operadors de més gran a més petita:

<i>operador</i>	<i>descripció</i>
()	parèntesis
-	mirall
*	intersecció i replicació
+ -	unió i desplaçaments

El llenguatge admet l'ús d'*identificadors* i d'*assignacions* mitjançant l'operador $:=$. Els

identificadors han de ser una lletra seguida de zero o més lletres o dígit.

Tasques

La vostra pràctica ha de contenir els components que es detallen a continuació.

Classe Skyline

Dissenyeu una classe `Skyline` per a la gestió dels *skylines*.

Cap operació pot tenir un cost quadràtic amb el nombre d'edificis. Per exemple, una configuració `{100000,20,3,1,10000}` no hauria de trigar més de 30 segons.

Gramàtica

Definiu la part lèxica (*tokens*) i sintàctica (gramàtica). Feu la gramàtica per a ANTLR4 . La regla inicial de la gramàtica ha de ser:

```
grammar Skyline;
```

Visitor

Definiu un *visitor* per a l'avaluació d'expressions del llenguatge descrit anteriorment.

Nota: caldrà utilitzar algun tipus de taula de símbols per gestionar les assignacions i els identificadors.

Telegram

La vostra tasca consisteix en implementar un Bot de Telegram que interactui textualment i gràficament amb els components que es detallen a continuació.

Noteu que les dades han de ser guardades per a cada usuari. El bot ha de funcionar amb diversos usuaris treballant al mateix temps amb dades independents entre ells.

Comandes del bot

El vostre Bot ha d'entendre i contestar correctament les comandes següents:

- `/start` inicia la conversa amb el Bot.

- `/help` el Bot ha de contestar amb una llista de totes les possibles comandes i una breu documentació sobre el seu propòsit i ús.
- `/author` el Bot ha d'escriure el nom complet de l'autor del projecte i seu correu electrònic oficial de la facultat.
- `/lst` : mostra els identificadors definits i la seva corresponent àrea.
- `/clean` : esborra tots els identificadors definits.
- `/save id` : ha de guardar un skyline definit amb el nom `id.sky` .
- `/load id` : ha de carregar un skyline de l'arxiu `id.sky` .

Intèrpret

Feu un intèrpret que interactui amb l'usuari com a l'exemple mostrat a l'inici del document, mitjançant el `MessageHandler` del `python-telegram-bot` .

Llibries

Utilitzeu les llibries de Python següents:

- `matplotlib` per graficar dades,
- `pickle` per guardar i carregar estructures de dades en binari, i
- `python-telegram-bot` per interactuar amb Telegram.

Podeu utilitzar lliurament les llibries estàndard de Python, però si no són estàndards, heu de demanar permís als vostres professors (que segurament no us el donaran).

Referències

- [Matplotlib](#). The Matplotlib development team, 2018.
- [pickle](#) — Python object serialization. Python Software Foundation, 2019.
- [Bots de Telegram](#). Jordi Petit, 2019.

Lliurament

Heu de lliurar la vostra pràctica al Racó. Només heu de lliurar un fitxer ZIP que, al descomprimir-se generi:

- un fitxer `requirements.txt` amb les llibreries que utilitza el vostre projecte (vegeu, per exemple, https://pip.pypa.io/en/stable/user_guide/#requirements-files),
- un fitxer `README.md` que el documenti (vegeu, per exemple, <https://gist.github.com/PurpleBooth/109311bb0361f32d87a2>),
- una carpeta `cl` amb el contingut de la part de compiladors (els arxius principals s'han de dir `test.*.py`),
- un arxiu `bot.py` amb la part del chatbot,
- un arxiu `skyline.py` amb la classe `Skyline`,
- un arxiu `document.pdf|png|jpeg` amb el [Compromís d'integritat acadèmica de la UPC](#) signat.

Els vostres fitxers de codi en Python han de seguir les regles d'estíl PEP8, tot i que podeu oblidar les restriccions sobre la llargada màxima de les línies. Podeu utilitzar el paquet `pep8` o <http://pep8online.com/> per assegurar-vos que seguïu aquestes regles d'estíl. L'ús de tabuladors en el codi queda prohibit (zero directe).

El termini de lliurament és el **dilluns 8 de juny a les 23:59 CEST**.

This site is open source. [Improve this page](#).