

Практикум из програмирања 1 (13Е111ПП1, 13С111ПП1)

- Четврти домаћи задатак -

Напомене:

- а)** Домаћи задатак се предаје преко **Moodle** курса предмета.
- б)** Рок за предају и термин одбране домаћег задатка ће бити благовремено објављени.
- в)** Домаћи задатак оцењује се путем јавних и тајних тестова за време предаје, израдом модификације оригиналног задатка за време одбране и усменим одговарањем за време одбране.
- г)** Домаћи задатак се решава самостално. Предметни наставници задржавају право да након предаје домаћег задатка изврше проверу сличности и предузму одговарајуће дисциплинске мере. Током израде решења није дозвољена употреба алата вештачке интелигенције заснованих на великим језичким моделима (*ChatGPT, Github Copilot* и сл.).
- д)** Програм мора да буде декомпонован на функције. Потпрограми не смеју да приступају променљивама главног програма директно, већ само путем својих аргумената и повратне вредности. Главни програм треба да позове функције које учитавају све потребне податке, затим да позове функције које врше захтевану обраду и да позове функцију за испис потребних резултата.

На програмском језику **Python** написати програм који ће, ради поједностављења разумевања израза написаног на неком програмском језику, паровима облик (малих) заграда обухватити изразе сачињене од операнада и оператора на основу броја њихових операнада, приоритета оператора и смера груписања. Оператори вишег приоритета имају предност у груписању операнада, док се у случају више оператора истог приоритета даје предност оном који групише са лева на десно, а у случају више таквих са истим смером груписања предност се даје оном оператору који се налази левље у изразу, уколико је њихов смер груписања са лева на десно, односно дешње у изразу, у супротном. Према броју операнада оператори могу бити унарни или бинарни. Смер груписања може бити L, што означава груписање са лева на десно или R, што означава груписање са десна на лево. У контексту унарних оператора, смер груписања односи се на место операнда на који се оператор примењује: L означава да се операнд налази лево од оператора, а R да се операнд налази десно од оператора. Сматрати да су изрази исправни, да су сви оператори израза претходно наведени и јединствени, као да су оператори и операнди израза раздвојени барем једним бланко знаком и да се у изразу не налази ниједна заграда.

Програм треба да:

1. Формира речник оператора тако што из прве линије стандардног улаза учита операторе и број операнада, приоритете, смер груписања, при чему је прва линија текста за сваки оператор задата у формату **оп1,брОперанада,приоритет,смерГруписања** који се могу јавити произвољан број пута у тој линији, при чему су у том случају раздвојени знаком ;.
2. Из друге линије стандардног улаза учита израз и уклони сва вишеструка узастопна појављивања бланко знакова тако да оператори и операнди буду раздвојени тачно једним бланко знаком. Након тога испише измењени израз на стандардни излаз.
3. Изврши процесирање оригиналног израза у складу са захтевом задатка. Најпре се формира листа стрингова коју чине појединачни операнди и оператори. Затим се бира оператор по претходно описаном критеријуму и везује се за операнд(е) чиме се формира подизраз типа стринга, а оператор и операнд(и) који чине тај подизраз избацују се из листе, док се новоформирани подизраз типа стринга додаје уместо њих на њихово место у листу. Поступак се понавља док год постоји могућност избора оператора, односно док листа не постигне величину 1 (новоформирани подизраз је, заправо, цео израз).
4. Испише резултујући новодобијени израз настао обрадом у претходном кораку.

Примери:

Стандардни улаз	Стандардни излаз
+,2,1,L;- ,2,1,L;* ,2,2,L;/ ,2,2,L a + b - c * d / e	a + b - c * d / e ((a + b) - ((c * d) / e))
+,2,1,R;- ,2,1,L;* ,2,2,L;/ ,2,2,L sum + inc - diff	sum + inc - diff (sum + (inc - diff))
#,1,4,R;+ ,2,2,L;- ,2,2,L;/ ,2,3,L;** ,2,4,L cnt + # diff - e ** num / den	cnt + # diff - e ** num / den ((cnt + (# diff)) - ((e ** num) / den))