

DU 1

Pokyny:

- Deadline na odovzdanie je 18.10.2023, 23:59:00.
- Odovzdava sa zdrojovy kod v ZIP archive do Teamsu.
- Svoj kod piste do baliku s nazvom `sat`.
- Piste citatelny a strukturovany kod, za spagety bude malo bodov!
- Dokopy mozete ziskat 10 bodov.
- V pripade lubovolnych nejasnosti piste, volajte, kricte.

Cast A: Definicia (4 body)

Pouzitim navrhoveho vzoru Composite definujte interface `Term` pre objekty, ktore budu zodpovedat nasledovnym vyrazom vo vyrokovej logike:

- Konstanta. Moze nadobudnut hodnoty `true` alebo `false`. Pravdivostnu hodnotu si bude drzat v sukromnej premennej typu `boolean`.
- Premenna. Tento typ v sebe nesie sukromnu premennu typu `String`, ktora oznacuje jej nazov.
- Implikacia. Klasicka implikacia medzi dvoma vyrazmi.
- And. Taktiez klasicka konjunkcia, avsak operandov moze byt viac ako 2. Operandy budu ulozene v (samozrejme sukromnom) ArrayList-e.
- Or. Analogicky k ANDu.

Interface `Term` by mal obsahovat metody:

- `public boolean Eval(Map<String, Bool> values)`, kde values je nejake ohodnotenie premennych.
- `public Term negate()`, ktora vrati vyraz zodpovedajuci negacii aktualneho vyrazu.
- `public ArrayList<String> getVariables()`, ktora vrati zoznam nazvov vsetkych premennych vyskytujucich sa vo vyraze (v lubovolnom poradi).

Cast B: toString (2 body)

Pridajte do Interfaceu `Term` funkciu `public String toString()`, ktora dany vyraz vypise v (pre cloveka) jednoducho citatelnom formate. Samozrejme treba dbat na spravne uzatvorkovanie pri zlozenych vyrazoch.

Cast C: SAT (3 body)

Pridajte do interfaceu `Term` funkciu `boolean isSatisfiable()`, ktora pre dany term zisti, ci existuje nejake ohodnotenie jeho premennych, pre ktore je vyraz pravdivy.

Cast D: Testy (1 bod)

Napiste si aspon 3 funkčne testy, napr. pomocou ``assert``-ov. Pozor, do VM treba pridať argument ``-ea`` pre správne fungovanie ``assert``-ov v kóde.