

Uvažujme tvrzení:

- (i) I když při prohledávání stavového prostoru do hloubky ošetříme detekci cyklů, nebude tento algoritmus úplný v obecných stavových prostorech (tedy včetně nekonečných stavových prostorů).
- (ii) Aby byl algoritmus  $A^*$  optimální, musí dodaná heuristika splňovat omezení přípustnosti  $\forall n: 0 \leq h(n) \leq h^*(n)$ , kde  $h^*(n)$  je skutečná cena cesty z  $n$  do cíle.
- (iii) Při prohledávání stavového prostoru se může více uzlů prohledávacího stromu odkazovat na tentýž stav.
- (iv) Algoritmus  $A^*$  expanduje všechny uzly, které mají  $f(n) < C^*$  ( $C^*$  je cena optimálního řešení) a některé uzly, které mají  $f(n) = C^*$ .
- (v) Cyklus se v Prologu standardně řeší pomocí rekurze.



- (A) Právě jedno z uvedených tvrzení je pravdivé.
- (B) Přesně dvě z uvedených tvrzení jsou pravdivá.
- (C) Přesně dvě z uvedených tvrzení jsou nepravdivá.
- (D) Právě jedno z uvedených tvrzení je nepravdivé.
- (E) Všechna uvedená tvrzení jsou pravdivá.

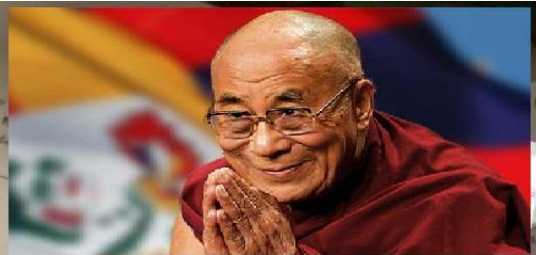
Máme definovaný predikát  $a/2$ , resp. funkci  $a(xs)$ , jako:

PL:

```

a([], []).
a([H/T], L):- a(T, V), b(H, V, L).
b(A, L, [A/L]).
b(A, [H/T1], [H/T2]):- b(A, T1, T2).

```



z láskou dla  
wszystkich studiujacych  
na tym uniwersytecie

PY:

```

def a(xs):
    if xs == Nil: yield Nil
    else:
        for ys in a(xs.tail):
            for zs in b(xs.head, ys):
                yield zs
def b(x, ys):
    yield Cons(x, ys)
    if not ys == Nil:
        for zs in b(x, ys.tail):
            yield Cons(ys.head, zs)

```

Co je prvním výsledkem volání  $?- a([x, y, y, z], L)$ , resp.  $L = a([x, y, y, z])$ ? Výsledky jsou uvedeny v prologovské notaci, pro Python platí ekvivalenty pomocí Nil/Cons.

$L = []$     $L = [z, y, y, x]$     $L = [y, y, z]$     $L = [x, y, y, z]$     $L = [x, z]$

Odpověď:

Které z následujících tvrzení pravdivě srovnává vlastnosti prohledávání podle ceny (Uniform-cost Search) a hledání nejlepší cesty  $A^*$ ?

Poznámka:  $d$  je hloubka cíle,  $b$  faktor větvení,  $f(n)$  je ohodnocovací funkce a  $C^*$  je skutečná cena optimálního řešení.

- (A) Algoritmus prohledávání podle ceny uspořádává uzly podle skutečné délky cesty do cíle přes daný uzel,  $A^*$  podle odhadu délky takové cesty.
- (B) Časová složitost obou algoritmů je exponenciální v závislosti na hloubce řešení  $d$ , tedy  $O(b^{d+1})$ .
- (C) Prostorová složitost obou algoritmů odpovídá počtu skutečně expandovaných uzlů s  $f(n) \leq C^*$ .
- (D) Oba algoritmy jsou optimální pro libovolné ocenění přechodů stavového prostoru.
- (E) Algoritmus prohledávání podle ceny vyžaduje pro úplnost kladné ohodnocení hran/cyklů,  $A^*$  tuto podmínku nemá.

