

Protože můžeme typy žetonů označovat jakkoliv, označuju je prvními písmeny, tím pádem písmeno *č* znamená červenou, *m* znamená modrou atd.

1 Část a

Důležité je si uvědomit, že červených žetonů musí být ostře více než modrých žetonů. Proto výsledný program bude:

```
č, m -> 2t
m -> t
č -> r
r, t -> 2r
```

Když je červených více, vzniklé tyrkysové žetony budou nahrazeny, jinak zůstanou tyrkysové žetony. To splňuje ostrou nerovnost v zadání. Zároveň nutně nemůžou červený a modrý zůstat.

2 Část b

Tady si pro pomoc zavedu podmínky $z \leq 1$ a $\check{z} \leq 1$. Zelený žeton zajistí, aby k příkazu inicializace (první příkaz) došlo jen jednou, následně použiju tmavě červené a tmavě modré žetony jako přechodné žetony a pak žlutý žeton nastartuje převod tmavě červených a tmavě modrých žetonů na červené, modré a fialové žetony. Kód programu je:

```
č, m -> tč, tm, z
tč, ž -> č, f, ž
tm, ž -> m, f, ž
č, tč, z -> 2tč, z
m, tm, z -> 2tm, z
z -> z, ž
```

Program se nezacyklí díky podmínkám výše.

3 Část c

Tady zelený žeton vznikne z červeného žetonu a slouží k odstartování kola, během kterého vzniknou tmavě modré žetony ze všech modrých a z nich pak vzniknou modré a fialové žetony. Během toho kola se zelený žeton odstraní a zároveň když už zůstanou jen fialové a modré žetony, modré žetony se všechny smažou, tím pádem zůstanou jen fialové žetony. Takže to funguje jako přičítání m c -krát.

```
m, z -> tm, z
tm, z -> m, f
tm -> m, f
č -> z
m ->
```