7. Házi feladat | Dokumentáció  
Magyar Tamás – 22 csoport – RNYR2F – [rnyr2f@inf.elte.hu](mailto:rnyr2f@inf.elte.hu) – 2020-04-6

Tartalom

[Feladat: 1](#_Toc37069295)

[Specifikáció: 1](#_Toc37069296)

[Megfeleltetés - Maximum kiválasztás: 2](#_Toc37069297)

[Algoritmus: 2](#_Toc37069298)

[Felsoroló: 2](#_Toc37069299)

[next() művelet: 2](#_Toc37069300)

[Megfeleltetés – Összegzés: 2](#_Toc37069301)

[Algoritmus: 2](#_Toc37069302)

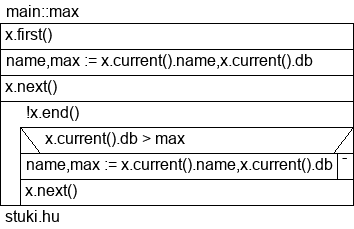
## Feladat:

Egy szekvenciális inputfájl ételreceptek hozzávalóit tartalmazza. A fájl egy eleme egy recept nevéből (sztring), és annak egyik hozzávalójának anyagnevéből (sztring), mennyiségéből (természetes szám) és mennyiségi egységéből (sztring) áll. A fájl nem üres, és recept nevek szerint rendezett, így ugyanazon recept hozzávalói közvetlenül egymás után helyezkednek el. Melyik ételrecepthez kell a legtöbb összetevő? (hozzávalók száma)

### Specifikáció:

A =   
(   
x:enor(RecipeLength),  
name: 𝕊,  
max:  
Recipe = rec(name:𝕊, ingredient:𝕊, amount:, amount\_type:𝕊)  
RecipeLength = rec(name:𝕊, db:)  
)

Ef = ( x = x’ ∧ x↗name ∧ |x|>0 )

Uf =( (name, max) =e.db)

### Megfeleltetés - Maximum kiválasztás:

t:enor(E) ~ x:enor(RecipeLength)  
f(e) ~ e.db  
H, < ~ N, <

### Algoritmus:

## Felsoroló:

t:enor(RecipeLength) RecipeLength = rec(name:𝕊, db:)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Order\*** | first() | next() | current() : RecipeLength | end() : 𝕃 |
| x : infile(Recipe) dx : Recipe  curr: RecipeLength sx : Status end : L | sx,dx,x:read  next() | lásd külön | **return** curr | **return** end |

### next() művelet:

A = (x : infile(Recipe), dx: Recipe, curr: RecipeLength, sx:Status, end:𝕃 )

Ef = ( x = x’ ∧ x↗name ∧ dx = dx’ ∧ sx = sx’ )

Uf =   
(  
end=(sx’=abnorm) ∧ (¬end → curr.name=dx’.name ∧

(curr.db, (sx,dx,x)) =   
)

### Megfeleltetés – Összegzés:

t:enor(E) ~ x:infile(Order) ) (sx,dx,x:read) first() nélkül, felt: dx.name = curr.name  
f(e) ~ 1   
s ~ curr.db  
H, +, 0 ~ , + , 0

### Algoritmus: