

## [2013/2014][Turek] Zbiór kartkówek z labów

<https://forum.iiet.pl/viewtopic.php?f=425&t=22428>

Strona 1 z 1

## [2013/2014][Turek] Zbiór kartkówek z labów

przez **ant6**

Napisane: **14 lutego 2014, o 00:11**

### Kartkówka 1

1. Jak uruchomisz funkcję 'stop' z modułu 'thread'?
2. Jak nazywa się w erlangu wbudowane makro zmieniające ścieżkę w powłoce? (o ile dobrze pamiętam, bo niestety nie wiedziałem ;<)
3. Jak sprawdzić, czy liczba n jest floatem? (is\_float(n).)
4. Jak usunąć wszystkie przypisania zmiennych? (f().)

### Kartkówka 2

1. Napisać funa w shellu, który przefiltruje listę tak, że zostaną tylko elementy podzielne przez coś tam i coś tam (głowy sobie uciąć nie dam, możliwe, że miało tylko sprawdzać podzielność dla liczby).
2. Jakie moduły będziemy dziś implementować? :)
3. list comprehension zwiększający każdy element listy o 3.
4. Zdefiniuj w terminalu erlanga (ale na kartce) funkcję, która zwraca prawdę dla liczb podzielnych przez 4 lub 7
5. Podaj nazwy modułów, które dzisiaj będziemy robić na zajęciach
6. Napisz funa sprawdzającego, czy liczba jest podzielna przez 5.
7. Napisz wyrażenie typu list comprehensions zwracające pary (x,x^2), gdzie x pochodzi z listy
8. fun sprawdzający, czy argument jest liczbą.

### Kartkówka 3

1. Komenda, po której proces wyśle do samego siebie ciąg znaków "ala ma kota",
2. Jak ma wyglądać moduł pFactorial? (Chodzi o te trzy warianty)
3. Jak tworzymy nowy proces i jak opróżniamy skrzynkę wiadomości.

### Kartkówka 4

1. Coś tam o przechwytywaniu sygnałów zakończenia //process\_flag(trap\_exit, true).
2. Wymienić wzorce projektowe otp (nie pamiętam, ile trzeba było podać. 4? 5? 666?). //Pewnie 5 bo tyle ich jest (application, supervisor, gen\_server, gen\_fsm i gen\_event).
3. Modele restartowania procesów we wzorcu supervisor z krótkim opisem (może być rysunkowy) - czyli one\_for\_one (restartowanie zakończonego procesu), one\_for\_all (ubicie i restart wszystkich z powodu zakończenia jednego), rest\_for\_one (ubicie i restart wszystkich zespawnowanych później niż ten, który się zakończył + ten proces)
4. Implementacja serwera mnożącego przez 4
5. Wypisz znane ci wzorce projektowe. (gen\_server, gen\_fsm, itd).

6. Jak zakończyć proces informując o tym proces rodzicielski?
7. Napisz serwer implementujący stos.
8. Napisz serwer przechowujący zmienna globalna, umożliwiającą zwiększanie/zmniejszanie jej (można było bez lub przy pomocy gen\_servera)