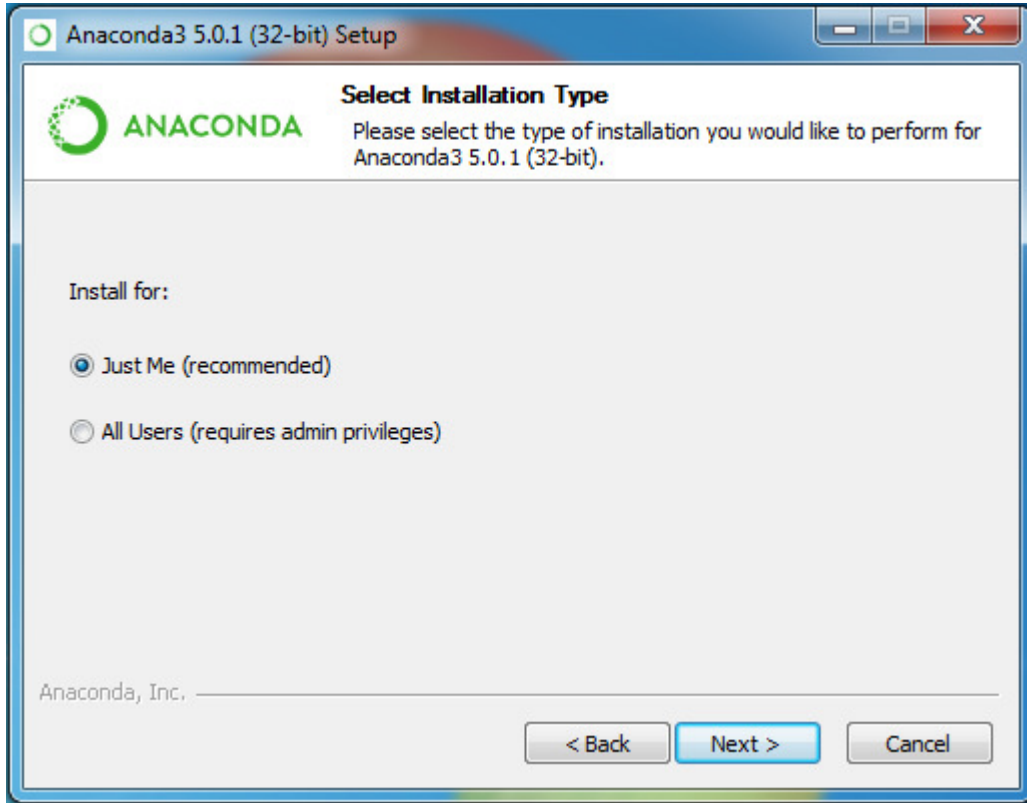


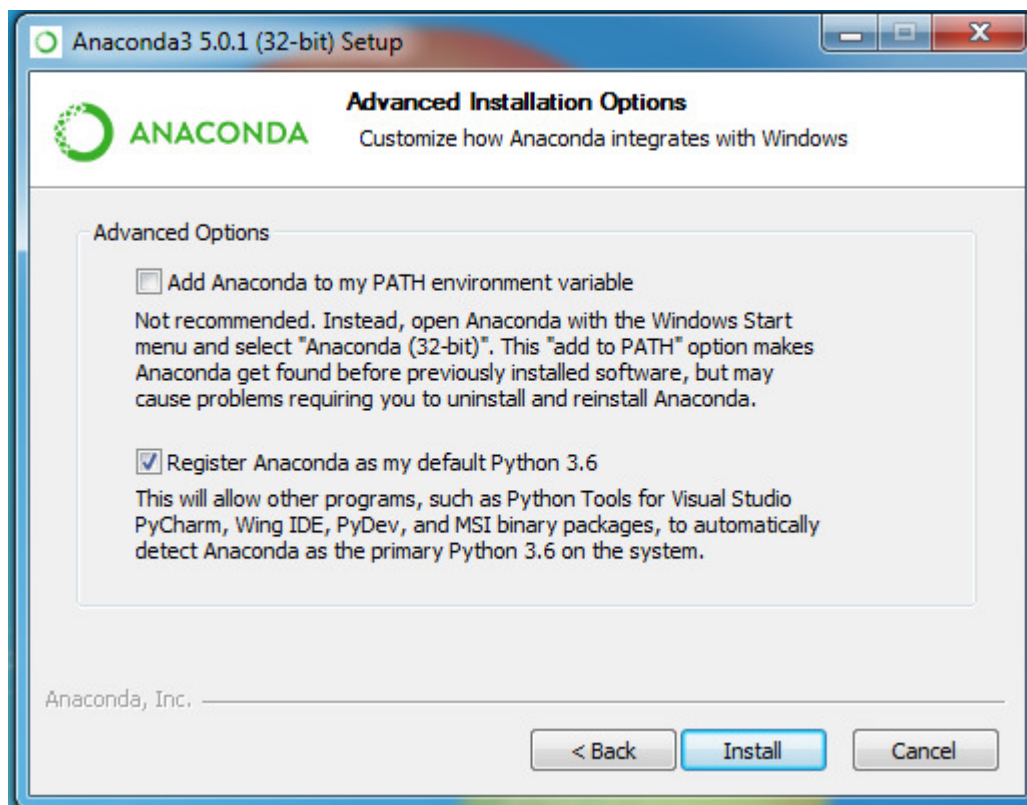
# Laborator 1

## Instalare Anaconda

1. Se descarca **Anaconda** (<https://www.anaconda.com/download/>); folositi versiunea pe 64 de biti daca sistemul de operare este de acelasi tip.
2. Pe parcursul instalarii se alege "Install just for me" - nu neceista drepturi de adminstrare pe calculatorul respectiv



3. Pentru destination folder alegeti o unitate si un director pentru care aveti drepturi de scriere, de ex. D:\lucru
4. Bifa de "Add Anaconda to my path environment variable" sa fie debifata, cea de "Register Anaconda..." bifata:



5. Actualizarea pachetelor se face in doi pasi:

## Setare mediu virtual, instalare de pachete

În Anaconda se pot defini medii virtuale (virtual environments). Avantajul acestora este că se permite ca într-un mediu să fie instalate pachete cu dependințe specifice fără a interfera cu pachetele din alte medii.

Implicit, la instalarea Anaconda se creează un mediu base. Se pot crea medii virtuale urmând pașii de mai jos.

1. Creați un mediu virtual (virtual environment) de data-science cu comanda:

```
conda create --name data-science python=3.6 anaconda --yes
```

Trecerea în mediul virtual se face cu:

```
activate data-science
```

iar ieșirea cu:

```
deactivate
```

Lista mediilor virtuale se obține prin:

```
conda info --envs
```

2. Instalați pachete auxiliare pentru Jupyter notebook:

```
conda install -c conda-forge jupyter_contrib_nbextensions  
jupyter contrib nbextension install --user
```

3. Se pot instala și pachete care nu se găsesc în canalul oficial de conda, folosind comanda pip. Orice distribuție de Python are acces la pachete prin acest manager:

```
pip install librosa
```

sau pentru upgrade al unui pachet instalat:

```
pip install keras --upgrade
```

Lista tuturor pachetelor instalate în distribuția Anaconda, în mediul curent, se obține cu comanda

```
conda list
```

Cele în dreptul cărora apare în ultima coloană specificația <pip> se pot actualiza doar prin package manager-ul pip; celelalte - folosind utilitarul conda.

## Manifestul Python

Lansați interpretorul interactiv Python în linia de comandă: `ipython` și apoi rulați comanda: `import this`. Se va afișa pe ecran manifestul Python the Zen of Python (<https://www.python.org/dev/peps/pep-0020/>).

## Ciclari, siruri de caractere

Se recomanda ca urmatoarele exercitii sa le lucrari in Jupyter notebook. In linia de comanda (command prompt sau PowerShell) scrieti comanda:

```
jupyter notebook
```

Se va deschide automa browserul implicit la adresa localhost:8888 (daca portul 8888 nu e liber, se va cauta automat un alt port).

1. (fizz-buzz test) Sa se scrie numerele de la 1 la  $n$ ; pentru fiecare multiplu de 3 se va scrie in locul numarului 'Fizz', pentru multiplu al lui 5 se va scrie 'Buzz'; daca numarul este multiplu de 15 se va scrie in locul lui 'FizzBuzz'.
2. Sa se verifice conjectura Collatz ([https://en.wikipedia.org/wiki/Collatz\\_conjecture](https://en.wikipedia.org/wiki/Collatz_conjecture)) pentru numerele intre 1 si 1000.
3. \* Sa se creeze o functie care preia un numar  $n$  si returneaza un alt numar pe baza cifrelor lui  $n$ , astfel: se calculeaza numarul cifrelor pare din  $n$  (fie el si 0), numarul de cifre impare (poate fi si 0), suma lor si se formeaza numarul din acestea 3; daca exista vreun zero nesemnificativ, acesta se va ignora. Numarul obtinut se va supune aceleiasi transformari. Exemplu: 3->11->22->202->303->123->123->123... Verificati ca dupa un numar finit de transformari se ajunge la numarul 123; faceti aceste verificari pentru numerele din intervalul 1, 1000.
4. \* Se pleaca de la un numar intreg. Fiecare cifra a sa se scrie cu litere, in limba engleza (de exemplu 5->five). Claculati numarul total de caractere rezultate, iar pentru numarul obtinut repetati procedura. Verificati pentru numerele de la 1 la  $n$  ca se obtine intr-un numar finit de pasi numarul 4. Exemplu: 123->onetwothree->11->oneone->6->six->3->three->5->five->4->four->4->four....
5. \* Este rezultatul de mai sus valabil si pentru transcriere in limba romana?
6. \* Se pleaca de la un numar  $n$ ; se scriu toti divizorii sai, inclusiv 1 si  $n$ ; se aduna *cifrele* tuturor acestor divizori; pentru numarul obtinut se aplica acelasi procedeu. Verificati ca procesul se stabilizeaza in numarul 15. Exemplu: 20->1. 2. 4. 5. 10. 20 -> suma cifrelor: 15->1. 3. 5. 15-> suma cifrelor: 15...