

C2184 Úvod do programování v Pythonu

Povinné domácí úkoly

Úkoly v této sadě se řeší opět do Jupyter notebooku. (Alternativně můžete psát řešení do `du_11.py`, pouze neměňte řádky začínající `#%%`. Testy si pak spustíte z příkazového řádku: `python testing.py du_11.py -s du_11_X`, kde X je číslo úkolu.)

Úkoly na sebe navazují, tj. v úkolech 11.2–11.5 nemusíte znovu psát načtení vstupních dat, ale můžete pracovat s proměnnými `years`, `temperatures` vytvořenými v úkolu 11.1. Také testy jsou udělané tak, že před spuštěním Vašeho řešení nastaví proměnné `years`, `temperatures` (plus `year_avg` v posledním úkolu).

Některé úkoly od Vás chtějí kromě výpisu i vykreslení grafu. Toto vykreslení se nijak netestuje (tj. dostanete body i bez správného vykreslení grafů). Berte to však jako přípravu na závěrečný test, kde se bude hodnotit i vykreslení grafu.

DÚ 11.1: Načtení dat

V souboru `data/temperatures.csv` máme uloženou tabulku průměrných měsíčních teplot z let 1961–2017.

Načtěte tato data do numpy pole. Vytvořte pole roků `years` a pole teplot `temperatures`.

Vypište:

- Tvar pole `years`, nejmenší a největší hodnotu v něm
- Pole `years`
- Tvar pole `temperatures`, nejmenší a největší hodnotu v něm
- Pole `temperatures`

Vzorový výstup (místo ... budou další řádky):

```
years: (57,) 1961 2017
[1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974
 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988
 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002
 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016
 2017]
temperatures: (57, 12) -8.2 24.0
```

```
[[-3.1  2.1  6.4 11.8 12.2 18.  16.7 17.9 16.5 10.9  3.8 -2.4]
 [-1.1 -1.3  0.2 10.4 11.5 15.5 16.8 18.8 12.9  8.9  3.8 -4.5]
 [-8.2 -6.1  0.8  9.9 14.1 17.6 19.8 18.5 15.4  8.5  6.8 -5.1]
 ...
 [-1.   4.9  5.7 10.2 15.6 19.8 21.3 19.6 18.3  9.1  4.  -0.3]
 [-5.1  1.6  8.3  9.2 16.  21.  21.2 22.1 14.1 10.6  4.6  1.9]]
```

Nápověda:

Použijte funkci `np.loadtxt` se správně nastaveným oddělovačem (viz dokumentace). První řádek může způsobovat problémy při načtení, protože má na začátku prázdnou buňku. Tento řádek ale obsahuje pouze čísla měsíců, takže ho můžeme klidně při načítání přeskočit (použitím vhodného parametru funkce `np.loadtxt` – najděte v dokumentaci).

Pro typovou konverzi se může hodit metoda `.astype`.

```
[ ]: import numpy as np
      from matplotlib import pyplot as plt

      data = ...
      years = ...
      temperatures = ...
```

DÚ 11.2: Přepočet jednotek

Vytvořte nové pole `temperatures_F`, ve kterém budou stejné teploty, ale přepočtené na °F (původní hodnoty jsou v °C).

Na přepočet lze použít jednoduchý vzorec:

$$T(^{\circ}\text{F}) = T(^{\circ}\text{C}) \times \frac{9}{5} + 32$$

Pak vypište:

- Tvar pole `temperatures_F`, nejmenší a největší hodnotu v něm
- Prvních 5 řádků z `temperatures_F` (tj. teploty za prvních 5 let)

Vzorový výstup:

```
temperatures [°F]: (57, 12) 17.240000000000002 75.2
[[26.42 35.78 43.52 53.24 53.96 64.4  62.06 64.22 61.7  51.62 38.84 27.68]
 [30.02 29.66 32.36 50.72 52.7  59.9  62.24 65.84 55.22 48.02 38.84 23.9 ]
 [17.24 21.02 33.44 49.82 57.38 63.68 67.64 65.3  59.72 47.3  44.24 22.82]
 [21.02 30.2  32.18 50.36 57.02 66.92 66.74 62.42 57.56 46.94 40.64 29.66]
 [30.02 25.88 36.5  45.86 53.42 61.88 62.78 60.62 58.28 44.78 33.26 32.36]]
```

```
[ ]: temperatures_F = ...

print('temperatures [°F]:', ...)
```

DÚ 11.3: Průměry

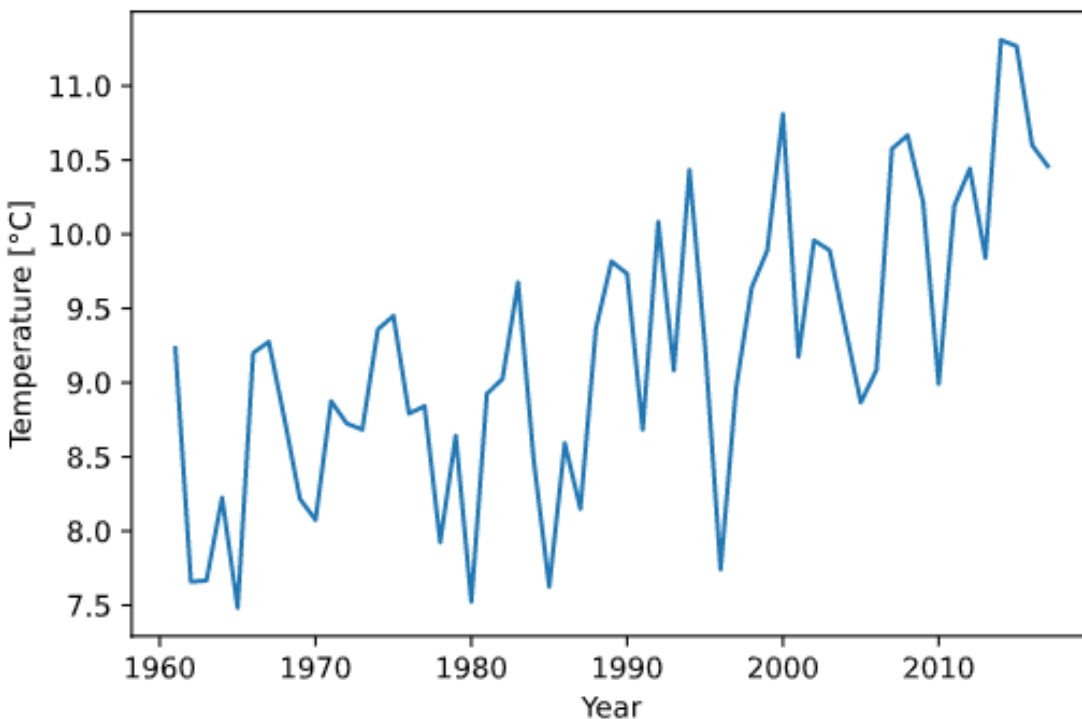
Spočítejte a vypište průměrné roční teploty `year_avg` a průměrné měsíční teploty za celé období `month_avg`.

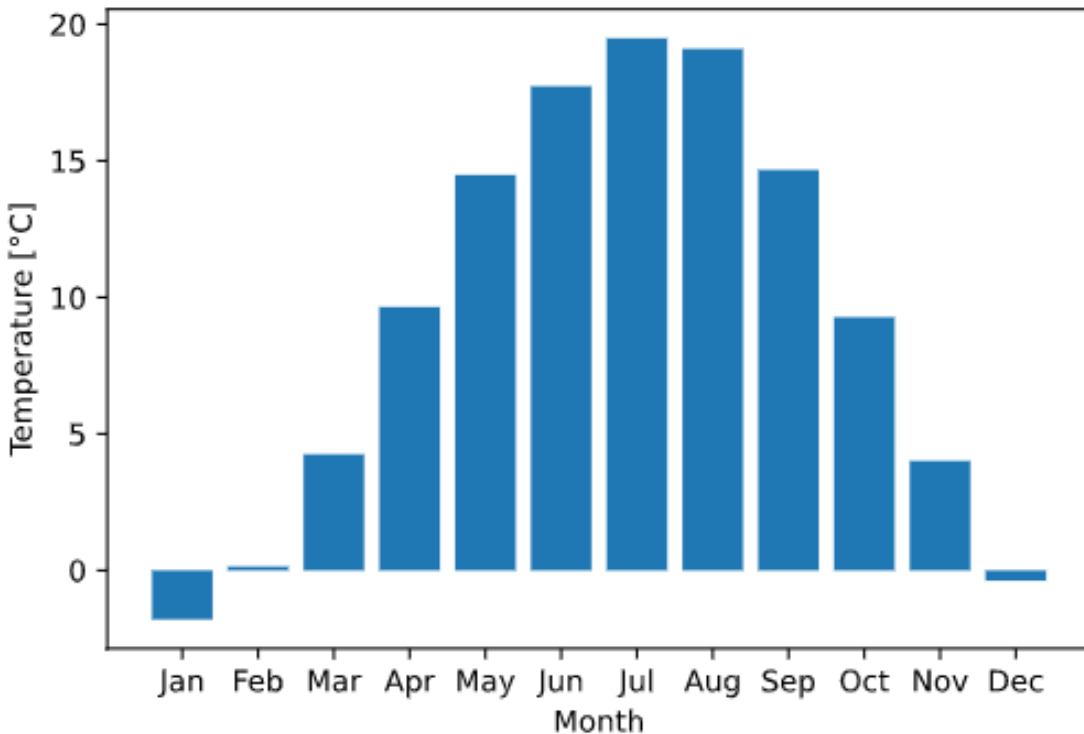
Pak tyto hodnoty vykreslete do grafů.

Vzorový výstup:

```
[ 9.23333333  7.65833333  7.66666667  8.225          7.48333333  9.2
  9.275         8.75833333  8.21666667  8.075          8.875         8.725
  8.68333333  9.35833333  9.45          8.79166667  8.84166667  7.925
  8.64166667  7.525         8.925         9.025          9.675         8.48333333
  7.625         8.59166667  8.15          9.36666667  9.81666667  9.73333333
  8.68333333  10.08333333  9.08333333  10.43333333  9.25          7.74166667
  8.96666667  9.64166667  9.89166667  10.80833333  9.175         9.95833333
  9.89166667  9.375         8.86666667  9.08333333  10.575        10.66666667
  10.21666667  8.99166667  10.19166667  10.44166667  9.84166667  11.30833333
  11.26666667  10.6         10.45833333]
[-1.79649123  0.14210526  4.25263158  9.64736842 14.48245614 17.73157895
 19.48947368 19.10175439 14.66842105  9.27368421  4.0122807  -0.3754386 ]
```

Vzorové grafy:





```
[ ]: year_avg = ...
    month_avg = ...

[ ]: plt.plot(years, year_avg)
    plt.xlabel('Year')
    plt.ylabel('Temperature [°C]')

[ ]: MONTHS = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec']
    plt.bar(MONTHS, month_avg)
    plt.xlabel('Month')
    plt.ylabel('Temperature [°C]')
```

DÚ 11.4: Teploty v lednu

Spočítejte a vypište:

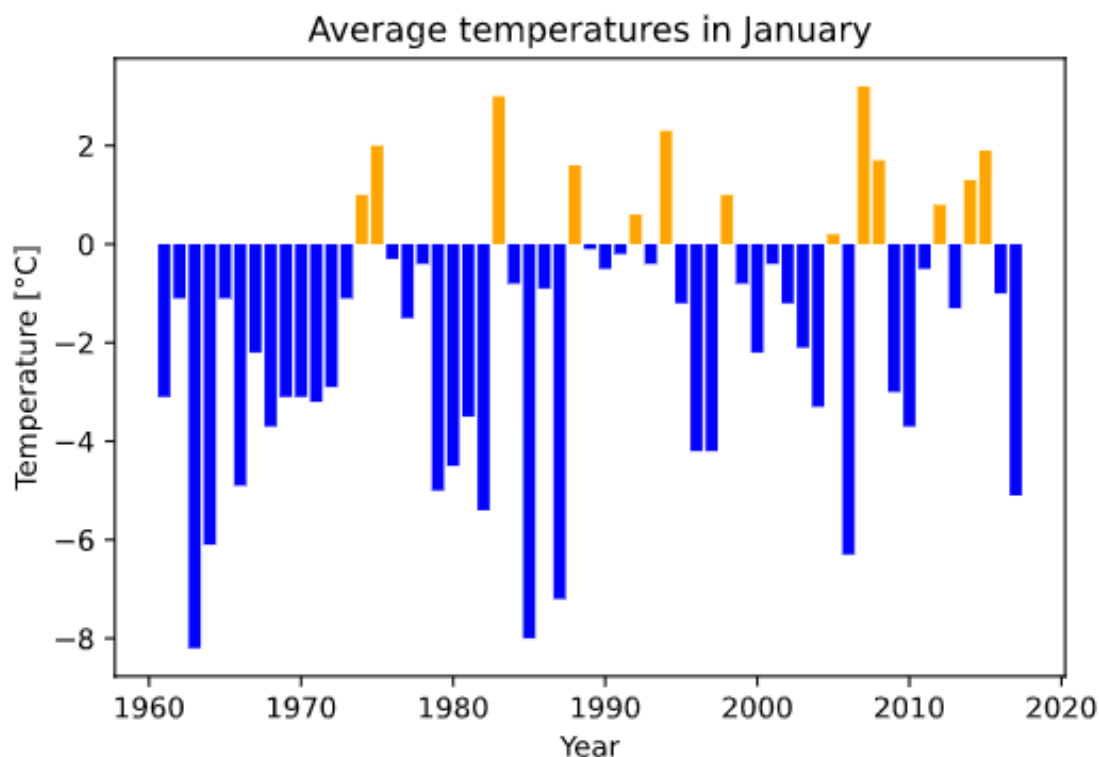
- Pole lednových teplot v jednotlivých letech.
- Masku kladných lednových teplot (tj. když je teplota nad nulou, v masce je True, když je teplota nula nebo pod nulou, v masce je False).
- Pole všech kladných lednových teplot (využijte již spočítanou masku).
- Pole všech záporných lednových teplot (včetně nulových, využijte již spočítanou masku).

Pak vytvořte sloupcový graf lednových teplot, kde kladné teploty budou barevne odlišeny (tip: lze to dosáhnout například tak, že funkci `plt.bar` zavoláme dvakrát – nejdřív vykreslíme pouze kladné, pak pouze záporné).

Vzorový výstup:

```
[ -3.1 -1.1 -8.2 -6.1 -1.1 -4.9 -2.2 -3.7 -3.1 -3.1 -3.2 -2.9 -1.1 1.
  2. -0.3 -1.5 -0.4 -5. -4.5 -3.5 -5.4 3. -0.8 -8. -0.9 -7.2 1.6
 -0.1 -0.5 -0.2 0.6 -0.4 2.3 -1.2 -4.2 -4.2 1. -0.8 -2.2 -0.4 -1.2
 -2.1 -3.3 0.2 -6.3 3.2 1.7 -3. -3.7 -0.5 0.8 -1.3 1.3 1.9 -1.
 -5.1]
[False False False False False False False False False False False False
 False True True False False False False False False False True False
 False False False True False False False True False True False False
 False True False False False False False False True False True True
 False False False True False True True False False]
[1. 2. 3. 1.6 0.6 2.3 1. 0.2 3.2 1.7 0.8 1.3 1.9]
[ -3.1 -1.1 -8.2 -6.1 -1.1 -4.9 -2.2 -3.7 -3.1 -3.1 -3.2 -2.9 -1.1 -0.3
 -1.5 -0.4 -5. -4.5 -3.5 -5.4 -0.8 -8. -0.9 -7.2 -0.1 -0.5 -0.2 -0.4
 -1.2 -4.2 -4.2 -0.8 -2.2 -0.4 -1.2 -2.1 -3.3 -6.3 -3. -3.7 -0.5 -1.3
 -1. -5.1]
```

Vzorový graf:



```
[ ]: january = ...
     mask = ...
```

DÚ 11.5: Odhad vývoje ročních teplot

V této úloze použijeme lineární regresi pro zjištění trendu v průměrných ročních teplotách.

To znamená, že reálná naměřená data (`year_avg`) proložíme přímkou, která bude mít rovnici:

$$T(x) = kx + q$$

kde $T(x)$ je odhadovaná teplota v roce x , a k a q jsou koeficienty přímky.

Použijte lineární regresi, funkce `np.polyfit`, pro zjištění koeficientů přímky. Pak z rovnice přímky odhadněte teplotu v roce 2030 ($T(2030)$).

Vypište koeficienty proložené přímky a odhad teploty v roce 2030 (zaokrouhlené na 3 desetinná místa).

Vykreslete do grafu reálné hodnoty a proloženou přímkou s odhadem do roku 2030.

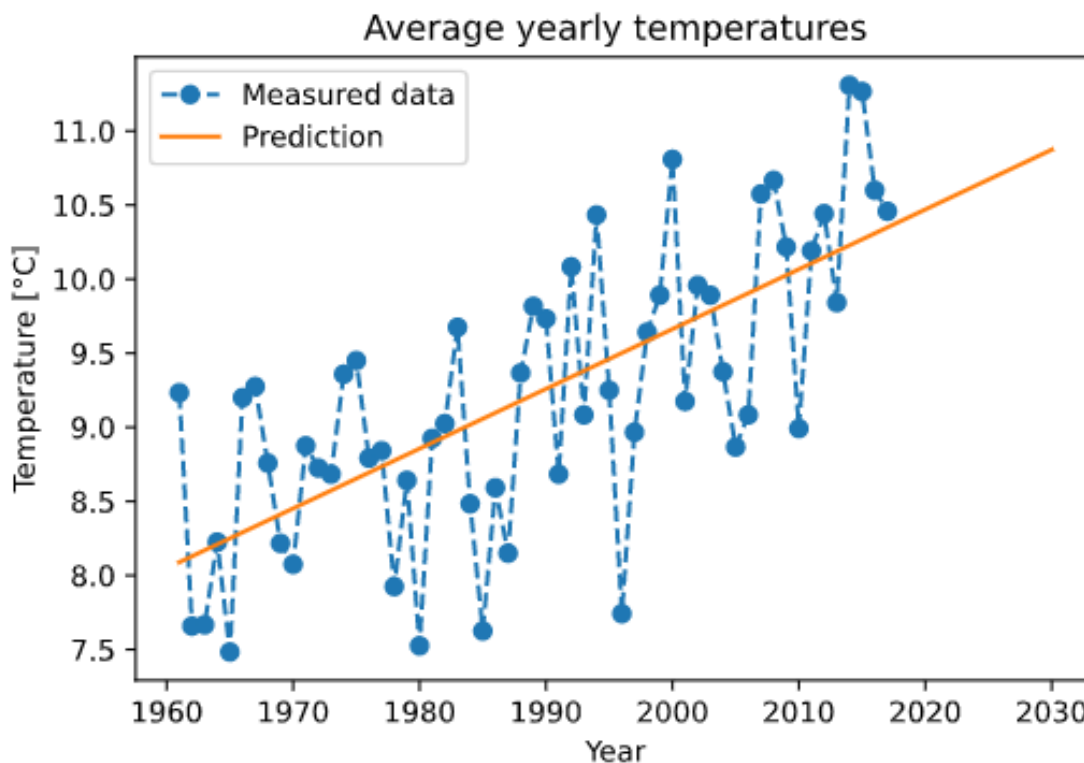
Vzorový výstup:

k: 0.040

q: -71.055

$T(2030)$: 10.874

Vzorový graf:



```
[ ]: k, q = ...  
t_2030 = ...
```