

1. (program: kalkulator_nizovi)

a) Napisati program u Pythonu koji za dva niza x i y:

```
x = [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
y = [2, 3, 4, 5, 6, 7, 8, 9, 10]
```

računa operacije: sabiranje, oduzimanje, množenje i deljenje odgovarajućih elemenata niza, kao i skalarni proizvod i ispisuje rezultate na ekran u obliku:

Zbir elemenata nizova x i y iznosi:

Razlika elemenata nizova x i y iznosi:

Proizvod elemenata nizova x i y iznosi:

Kolicnik elemenata nizova x i y iznosi:

Skalarni proizvod nizova x i y iznosi:

b) Na osnovu nizova x i y generisati kvadratne matrice a i b, respektivno, dimenzija 3 x 3:

```
Matrica_a = [[1 2 3]
```

```
             [4 5 6]
```

```
             [7 8 9]]
```

```
Matrica_b = [[2 3 4]
```

```
             [5 6 7]
```

```
             [8 9 10]]
```

pa nad njima izračunati operacije: sabiranje, oduzimanje, množenje i deljenje odgovarajućih elemenata matrica, kao i proizvod dve matrice a i b. Rezultate ispisi na ekran u obliku:

Zbir elemenata matrica a i b iznosi:

Razlika elemenata matrica a i b iznosi:

Proizvod elemenata matrica a i b iznosi:

Kolicnik elemenata matrica a i b iznosi:

Proizvod matrica a i b iznosi:

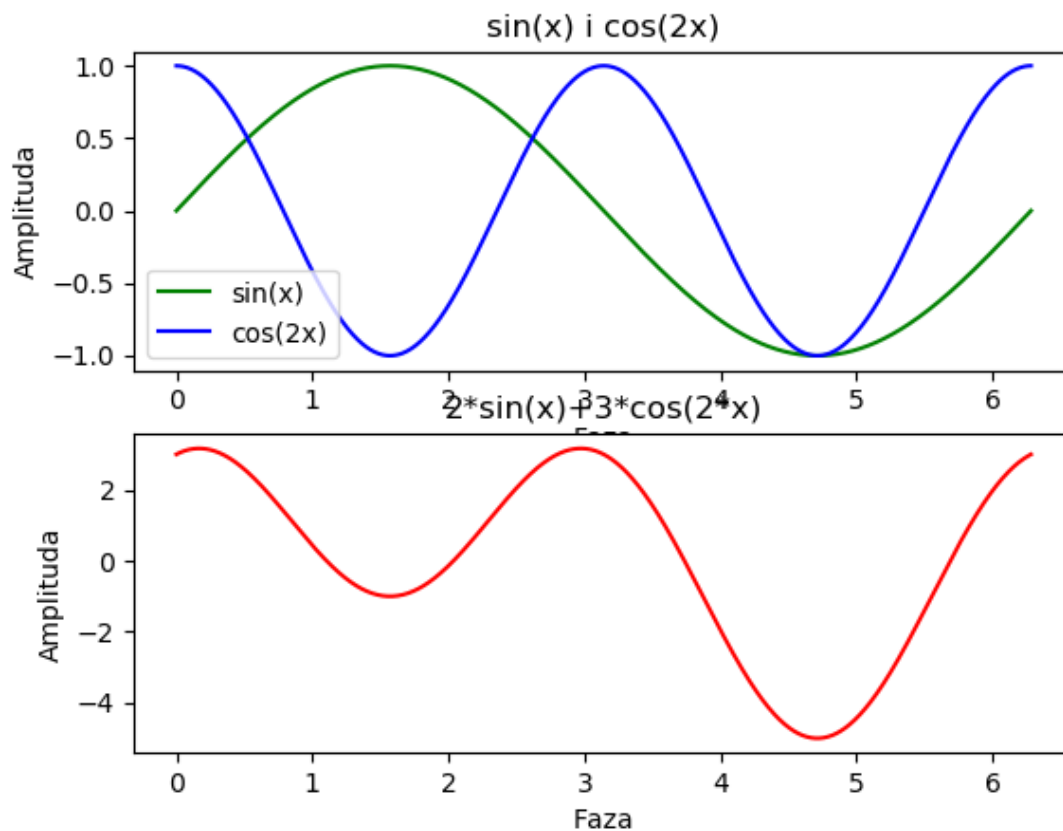
Napomena: koristiti funkcije **arrange()**, **reshape()**, **dot()**.

2. (program: signali)

Napisati funkciju u Pythonu koji crta sliku koja se sastoji iz dva dela (prikaz dat ispod):

- na prvom delu (gornjem) prikazati signale $\sin(\phi)$ i $\cos(2\phi)$. Signal $\sin(\phi)$ crtati linijom zelene a signal $\cos(2\phi)$ linijom plave boje. Ova slika treba da ima i legendu koja objasjava koji je koji signal.
- na drugom delu (donjem) prikazati signal: $2\sin(\phi) + 3\cos(2\phi)$ i nacrtati ga linijom crvene boje.

Napomena: Phi je faza signala i kreće se u rasponu od 0 do 2π , sa rezolucijom od 1000 tačaka unutar ovog intervala. Obe slike treba da imaju oznaku koordinatnih osa (x osa je Faza a y osa je Amplituda), kao i odgovarajuće naslove. Sliku je potrebno snimiti u dva formata, kao .png i .pdf. pod nazivom "Slika1". Koristiti funkciju **linspace()**, **subplot()**.



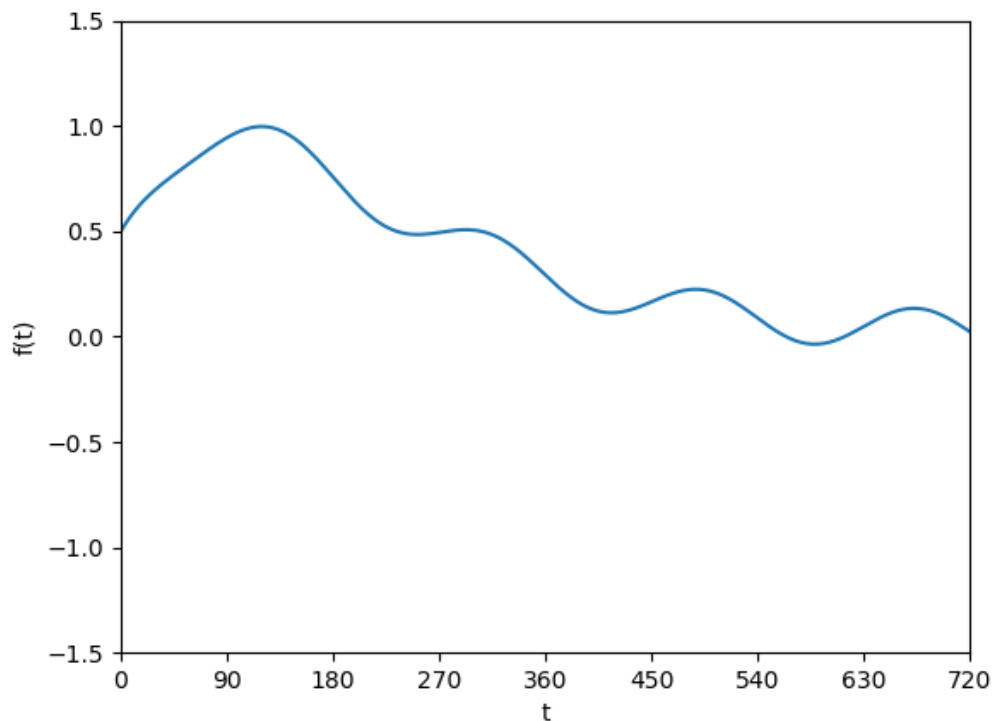
3. (program: primer_funkcije)

Napisati program u Pythonu koji generiše funkciju:

$$f(t) = (t + 0.5) * \exp(-0.5 * t) - 0.1 * \sin(2 * t)$$

i računa njene vrednosti u opsegu t od 0 do $4 * \pi$, u 721 tački. Ove vrednosti potrebno je upisati u datoteku "funkcija.txt" u formi dve kolone, od kojih prvu čine vrednosti promenljive t a drugu vrednosti funkcije f(t) za odgovarajuće vrednosti promenljive t.

Takođe, potrebno je nacrtati funkciju f(t), ali kada je t izraženo u stepenima (koristiti funkciju **degrees()** za konverziju radijana u stepene). Slika treba da ima oznake koordinatnih osa i granice po koordinatnim osama kao što je prikazano na slici ispod. Sliku sačuvati kao "funkcija.pdf".



4. (program: primer_matrice)

Napisati program u Pythonu koji definiše veličinu kvadratne matrice i njen sadržaj, tj. elemente (oba unosi korisnik). Elementi matrice su tipa **int**. Matricu zatim treba upisati u datoteku "matrica.txt" (širina polja za element matrice je 5 mesta). Zatim je potrebno izračunati njenu inverznu matricu i upisati je u datotetku "inverzna matrica.txt" (element je tipa float u formatu ukupne širine 14 mesta, i na 5 decimala). Program računa još i zbir elemenata matrice po vrstama i to takodje upisuje u treću datoteku, "zbir po vrstama.txt", u obliku:

zbir vrste 0 je:

zbir vrste 1 je:

.
. .
.