დედამიწა კოსმოსიდან (გადაღებულია **Apollo** 17 კოსმოსური ხომალდიდან)

4 MATLAB-ის გრაფიკული შესაძლებლობები

პრობლემა: გლობალური ცვლილების პროგნოზი

ბიოსფერო ის გარემოა, სადაც სიცოცხლე არსებობს. დედამიწის ბიოსფერო მოიცავს ატმოსფეროს, ჰიდროსფეროს – ზღვებსა და ოკეანეებს და ლითოსფეროს – დედამიწის ქერქის ნაწილს. იმისათვის რომ შევისწავლოთ ბიოსფეროში მიმდინარე ფიზიკური პროცესები, კარგად უნდა გავერკვეთ უამრავ წვრილმან დეტალში, როგორიცა მაგალითად, ნახშირორჟანგის მიმოქცევა ატმოსფეროსა და ოკეანეში, ოზონის საფარის შესუსტება, ქიმიური და ენერგეტიკული პროცესებით გამოწვეული კლიმატური ცვლილებები. საჭირო მინაცემების შესაგროვებლად გამოიყენება მეტეოროლოგიური რაკეტა, რომლის საშუალებით ხერხდება დედამიწის ატმოსფროს ზედა ფენების კვლევა. ატმოსფეროს სხადასხვა ფენის გავლისას რაკეტის ცხვირზე დამაგრებული ტელემეტრული სისტემა გადმოსცემს მონაცემებს დედამიწაზე. დაგროვილი მონაცემები შემდგომში კომპიუტერის საშუალებით მუშავდება.

შესავალი

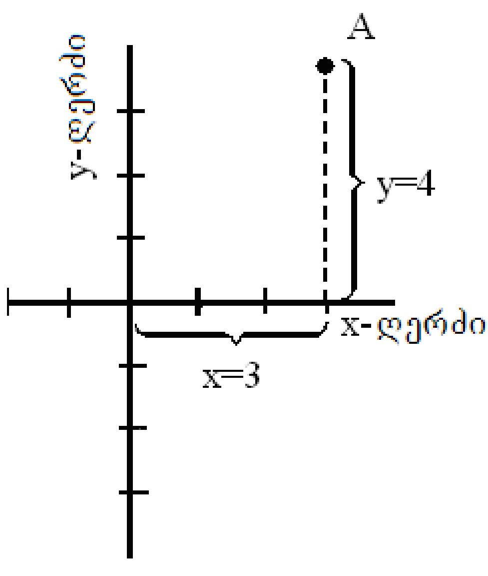
გრაფიკის აგება საშუალებას გვაძლევს შევამოწმოთ და გავაანალიზოთ მონაცემები. MATLAB -ს გაჩნია მძლავრი გრაფიკული საშუალებები. თუ შეისწავლით მათ, საშუალება გექნებათ მარტივად ააგოთ სხვადასხვა ფორმატის გრაფიკი, როგორიცაა x – y გრაფიკი, პოლარული, Bar და კონტურული, სამგანზომილებიანი გრაფიკი. ამ თავში განვიხილავთ MATLAB –ის გრაფიკულ ბრძანებებს და მათ მრავალფეროვან შესაძლებლობებს

4.1 X –Y გრაფიკი

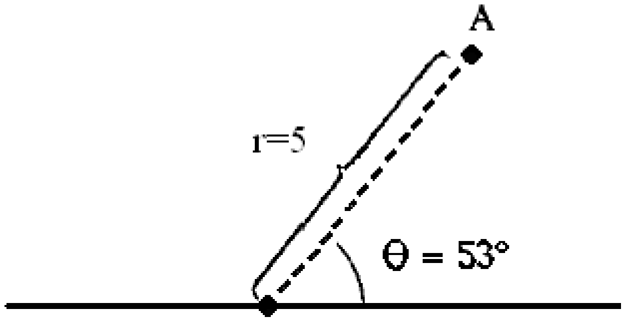
განვიხილოთ X – Y გრაფიკი, რომელიც ყველაზე ხშირად გამოიყენება. მონაცემები, რომელთა მიხედვითაც გრაფიკს ვაგებთ, ჩვეულბრივ მონაცემთა ფაილის სახითაა წარმოდგენილი ან მიიღება კომპიუტერული პროგრამით გათვალისწინებული გამოთვლებით, რომელიც იძლევა შესაბამის x და y ვექტორებს. საზოგადოდ ვთვლით, რომ x შეესაბამება დამოუკიდებელ ცვლადს, ხოლო y მასზე დამოკიდებულ ცვლადს. y მნიშვნელობები შესაძლოა გამოთვლილი იქნას როგორც x –ის ფუნქცია. შესაძლოა ორივე სიდიდე მიღებული იყოს ექსპერიმენტის შედეგად.

კოორდინატთა მართკუთხა სისტემა (დეკარტის)

სხვადასხვა ამოცანებში ვიყენებთ მონაცემებს რომელთა წარმოდგენა შესაძლებელია როგორც კოორდინატთა მართკუთხა, ისე პოლარულ სისტემაში. კოორდინატთა მართკუთხა სისტემაში წერტილის ადგილი განისაზღვრება მანძილით სათავიდან ჰორიზონტალური და ვერტიკალური ღერძების გასწვრივ ნახ 7.1. გრაფიკის ასაგებად კოორდინატთა პოლარულ სისტემაში გვჭირდება პოლარული მონაცემები – მანძილი კოორდინატთა სათავიდან და კუთხე ნახ 7.2. ჯერ განვიხილოთ გრაფიკის აგება კოორდინატთა მართკუთხა, დეკარტის სისტემაში, შემდეგ პოლარულ სისტემაში. ვიდრე წარმოგიდგენთ ბრძანებებს გრაფიკის ასაგებად, მიმოვიხილოთ უკვე განხილული ბრძანებები რომლებიც ნახაზს უკეთებს სათაურს და ღერძებს შესაბამის წარწერას დაურთავს. გრაფიკს დაუსრულებელი სახე აქვს თუ არ შეიცავს ამგვარ ინფორმაციას. ხშირად გრაფიკს დავურთავთ საკოორდინატო ბადეს, რადგან იგი აადვილებს იმ სიდიდეთა შეფასებას, რომლის მიხედვითაც იგი აიგება.



ნახ 7.1. კოორდინატთა მართკუთხა - დეკარტის სისტემა



ნახ 7.2. კოორდინატთა პოლარული სისტემა

Labels (წარწერა ღერძებზე)

title('text')	გრაფიკს გაუკეთებს სათაურს
xlabel('text')	x-ღერძის ქვემოთ წააწერს ბრჭყალებში მოთავსებულ ტექსტს
ylabel	y-ღერძის გასწვრივ წააწერს ბრჭყალებში მოთავსებულ ტექსტს
text(x,y,'text')	ბრჭყალებში ჩაწერილ ტექსტს წააწერს გრაფიკს წერტილში, რომლის კოორდინატებია x,y მოცემული გრაფიკის ღერძების მიხედვით, თუ x ,y ვექტორებია, წარწერას გააკეთებს ყოველ წერტილში
text(x,y, 'text','sc')	ბრჭყალებში ჩაწერილ ტექსტს წააწერს გრაფიკს წერტილში, რომლის კოორდინატებია x,y ისე, რომ ქვედა მარცხენა კუთხის კოორდინატებია (0,0), ზედა მარჯვენასი კი (1,1)
gtext('text')	წააწერს ტექსტს გრაფიკზე იმ წერტილში, რომელსაც მაუზით ან კლავიატურის ისრებით მივუთითებთ
grid	დაიტანს გრაფიკზე საკოორდინატო ბადეს

4.1.1 ბრძანება **plot**

სზოგადოდ როცა ამ ბრძანებით ვაგებთ გრაფიკს, იგულისხმება, რომ x და y ღერძები დაყოფილია თანატოლ ინტრვალებად. ასეთ გრაფიკულ ოპერაციას წრფივს უწოდებენ. ზოგჯერ შეიძლება საჭირო იყოს ლოგარითმული გრადაცია ერთ-ერთ ან ორივე ღერძზე. 10 ფუძიანი ლოგარითმული გრადაცია მოხერხებულია, როცა ცვლადის მნიშვნელობები ძალიან ფართო დიაპაზონშია განსაზღვრული.
MATLAB ბრძანებები წრფივი და ლოგარითმული გრადაციის გრაფიკებისათვის შემდეგია:

plot(x,y)	აგებს x,y სიდიდების წრფივ გრაფიკს x – დამოუკიდებელი ცვლადია, y მასზე დამოკიებული
semilogx(x,y)	აგებს გრაფიკს ლოგარითმული გრადაციით x –თვის და წრფივით y –თვის.
semilogy(,y)	აგებს გრაფიკს ლოგარითმული გრადაციით y –თვის და წრფივით x –თვის.
loglog(x,y)	აგებს x,y გრაფიკს ლოგარითმული გრადაციით ორივე

ცვლადიათვის

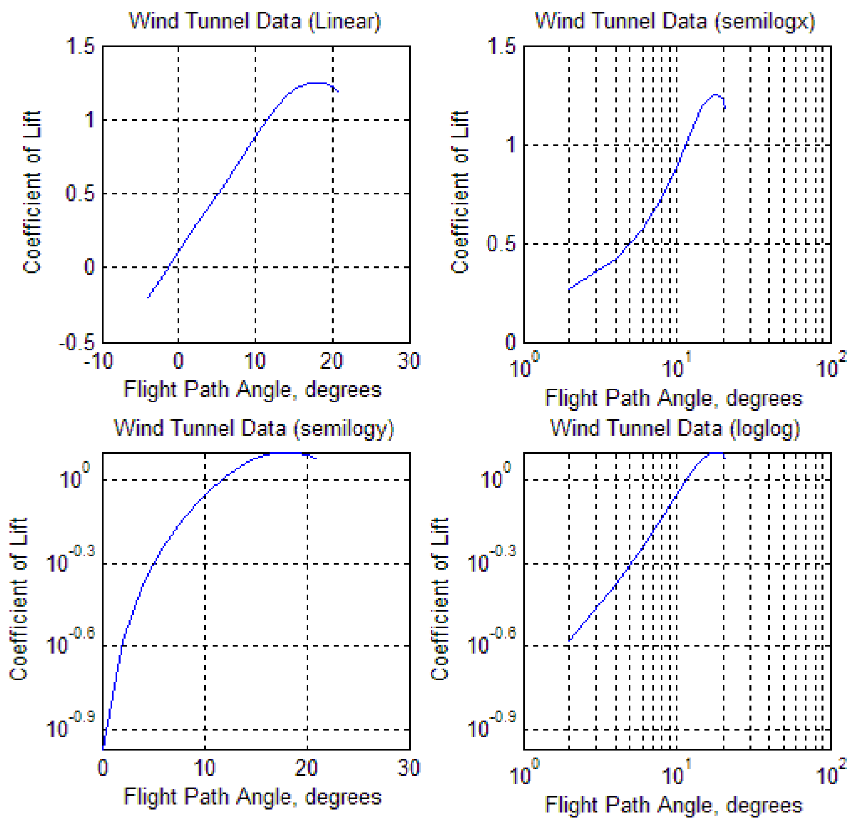
მნიშვნელოვანია გავიხსენოთ, რომ ნულის ტოლი ან ნულზე ნაკლები სიდიდის ლოგარითმი არ არსებობს, ამიტომ თუ მონაცემები, რომლებიც უნდა ავაგოთ **semilog** ან **loglog** ბრძანებით, შეიცავს უარყოფით ან ნულის ტოლ მნიშვნელობებს, MATLAB მიგვითითებს შეცდომაზე და გვამცნობს, რომ ეს მნიშვნელობები გამოტოვებული იქნება გრაფიკის აგების დროს.

თუ x ან y გრაფიკულ ბრძანებებში მატრიცაა, მაშინ გრაფიკი აიგება მატრიცის თითოეული სტრიქონის ან სვეტის მიმართ ერთიდაიგივე გრაფიკზე.

თუ x და y ერთნაირი ზომის მატრიცებია, აიგება x მატრიცის ყოველი სვეტი y შესაბამისი სვეტის მიმართ.

თითოეულ ამ ბრძანებათაგანს შეიძლება ჰქონდეს ერთი არგუმენტი. ამ შემთხვევაში x ვექტორის მნიშვნელობები აიგება ისეთი y ვექტორის მიმართ, რომლის ელემენტებია x მნიშვნელობათა შესაბამისი ინდექსი.

გავიხსენოთ, რომ მეორე თავში ავაგეთ თვითმფრინავის ფრენის ტრაექტორიის მნიშვნელობები lift (ამწევი ძალის) კოეფიციენტის მიმართ. ნახ 7.3 –ზე წარმოდგენილია ამ მნიშვნელობათა გრაფიკები წრფივი, semilogx, semilogy და loglog ბრძანებებით:



ნახ 7.3. წრფივი და ლოგარითმული გრაფიკები

პირველი მათგანი წარმოადგენს წრფივ გრაფიკს:

```
plot(x,y),...
title('Wind Tunnel Data (Linear)'),...
xlabel('Flight Path Angle, degrees'),...
ylabel('Coefficient of Lift'),...
grid
```

გავიხსენოთ, რომ x ვექტორის რამდენიმე მნიშვნელობა უარყოფითია, ამიტომ MATLAB ბრძანებათა ფანჯარაში ლოგარითმული გრადაციის შემთხვევაში მივიღებთ შესაბამის შეტყობინებას:

```
Warning: Negative data ignored.
```

იმისათვის, რომ უფრო თვალსაჩინო გრაფიკი შევარჩიოთ სასურველია ავაგოთ ჩვენი მონაცები სხვადასხვა ბრძანებათა გამოყენებით.

სავარჯიშო

შეადგინეთ 100 ელემენტიანი x ვექტორი 0 დან ბიჯით 0,5 და გამოითვალეთ შესაბამისი y როგორც x –ის ფუნქცია:

$$y = 5x^2$$

1. ააგეთ ამ მონაცემთა წრფივი გრაფიკი
2. ააგეთ ამ მონაცემთა გრაფიკი ლოგარითმული გრადაციით x-ის მიმართ
3. ააგეთ ამ მონაცემთა გრაფიკი ლოგარითმული გრადაციით y-ის მიმართ
4. ააგეთ ამ მონაცემთა loglog გრაფიკი
5. შეადარეთ გრაფიკები ერთმანეთს აღწერეთ თითოეულის უპირატესობა და ნაკლი

4.2 პოლარული გრაფიკი

ზოგჯერ მოცემული გვაქვს პოლარული კოორდინატები – სიდიდე და შესაბამისი კუთხე, მაგალითად ვზომავთ სინათლის ინტენსივობას წყაროს ირგვლივ. შეგვიძლია ინფორმაცია წარმოვადგინოთ როგორც რაიმე ფიქსირებული ღერძის მიმართ ათვლილი კუთხე და ინტენსივობა ამ მიმართულებით. ასეთი მონაცემები უფრო თვალსაჩინო სურათს მოგვცემს თუ ავაგებთ პოლარულ გრაფიკს. პოლარული გრაფიკი ასევე თვალსაჩინოდ წარმოგვიდგენს კომპლექსურ რიცხვებს.

4.2.1 პოლარული კოორდინატები

წერტილი კოორდინატთა პოლარულ სისტემაში განსაზღვრულია ორი სიდიდით – კუთხე θ და მოდული. კუთხის მნიშვნელობები ხშირად მოცემულია როგორც სიდიდეები 0 და 2π რადიანს შორის ($0 - 360^0$). მოდული დადებითი რიცხვია და წარმოადგენს მანძილს კოორდინატთა სათავიდან ამ წერტილამდე მოცემული მიმართულებით.

4.2.2 ბრძანება **polar**

MATLAB –ში პოლარული გრაფიკი აიგება ბრძანებით **polar**. მისი არგუმენტებია კუთხე (θ) და მოდული (r), მანძილი.

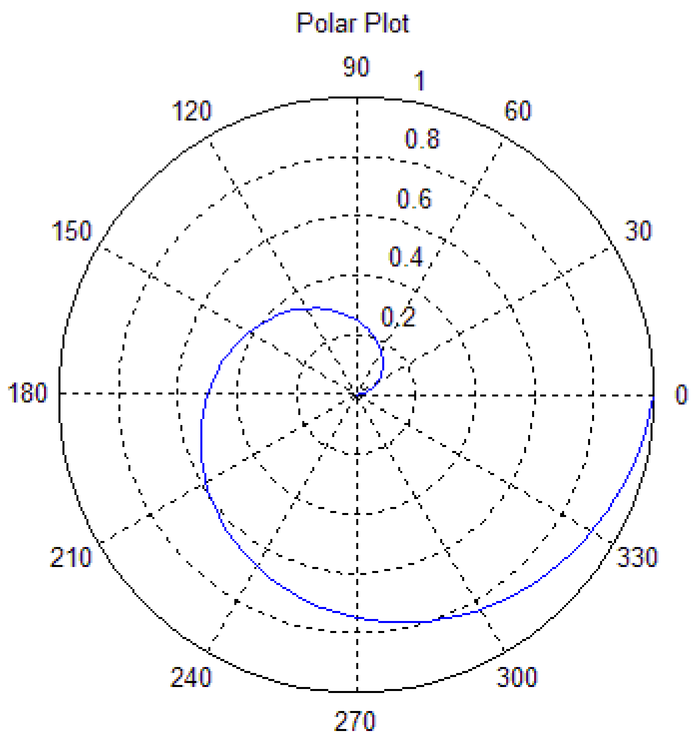
polar(theta,r)	ეს ბრძანება ააგებს პოლარულ გრაფიკს θ კუთხისა და შესაბამისი მოდულისათვის-r

თუ რომელიმე არგუმენტთაგანი მატრიცაა, მაშინ ვექტორი აიგება მატრიცის სვეტების ან სტრიქონების მიმართ, ერთდროულად მივიღებთ იმდენ მრუდს, რამდენი სტრიქონიც (სვეტიც) შედის მატრიცაში.

თუ ორივე არგუმენტი ერთნაირი ზომის მატრიცაა, ერთი მათგანის სვეტები აიგება მეორის შესაბამისი სვეტების მიმართ.

დავუშვათ გვინდა ავაგოთ მრუდი წრეწირის წერტილებისა ზრდადი რადიუსით. შევქმნით ვექტორს θ, რომლის ელემენტებია რადიანებში გამოსახული კუთხე 0 დან 2π-მდე. შევქმნათ ვექტორი r, რომლის ელემენტები იქნებს რადიუსების შესაბამისი მნიშვნელობები 0 დან 1-მდე. ნახ 7.4 შეიცავს გრაფიკს, რომელიც აიგება ბრძანებათა შემდეგი მწკრივით:

```
theta = 0:2*pi/100:2*pi;
r = theta/(2*pi);
polar(theta,r),...
    title('Polar Plot'),...
    grid
```



ნახ 7.4. პოლარული გრაფიკი წრე ზრდადი რადიუსით

4.3 მართკუთხა/პოლარული გარდაქმნა

ხშირად საჭიროა მონაცემები კოორდინატთა ერთი სისტემიდან მეორეში გადავიყვანოთ. ტრიგონომეტრიის გამოყენებით ასეთი ტრანსფორმაცია გამოისახება ფორმულებით:

პოლარული კოორდინატების გადაყვანა მართკუთხა სისემაში:

$$x = r\sin\theta$$
$$y = r\cos\theta$$

მართკუთხა კოორდინატების გადაყვანა პოლარულ სისტემაში:

$$r = \sqrt{x^2 + y^2}$$

$$\theta = \tan^{-1}\left(\frac{y}{x}\right)$$

ტანგენსის შებრუნებული სიდიდის გამოთვლისას სიფრთხილეა საჭირო, რომ სწორად შევარჩიოთ კუთხის მნიშვნელობა. თუ MATLAB-ს იყენებთ ამ გარდაქმნის განსახორციელებლად, შეგიძლიათ თავი დაიზღვიოთ **atan2** ფუნქციის გამოყენებით.

სავარჯიშო

გდაიყვანეთ მოცემული წერტილები კოორდინატთა მართკუთხადან პოლარულ სისტემაში:

1. (3, -2)
2. (0, -1)
3. (-2,0)
4. (0.5, 1)

გადაიყვანეთ შემდეგი წერტილები პოლარულიდან მართკუთხა სისტემაში:

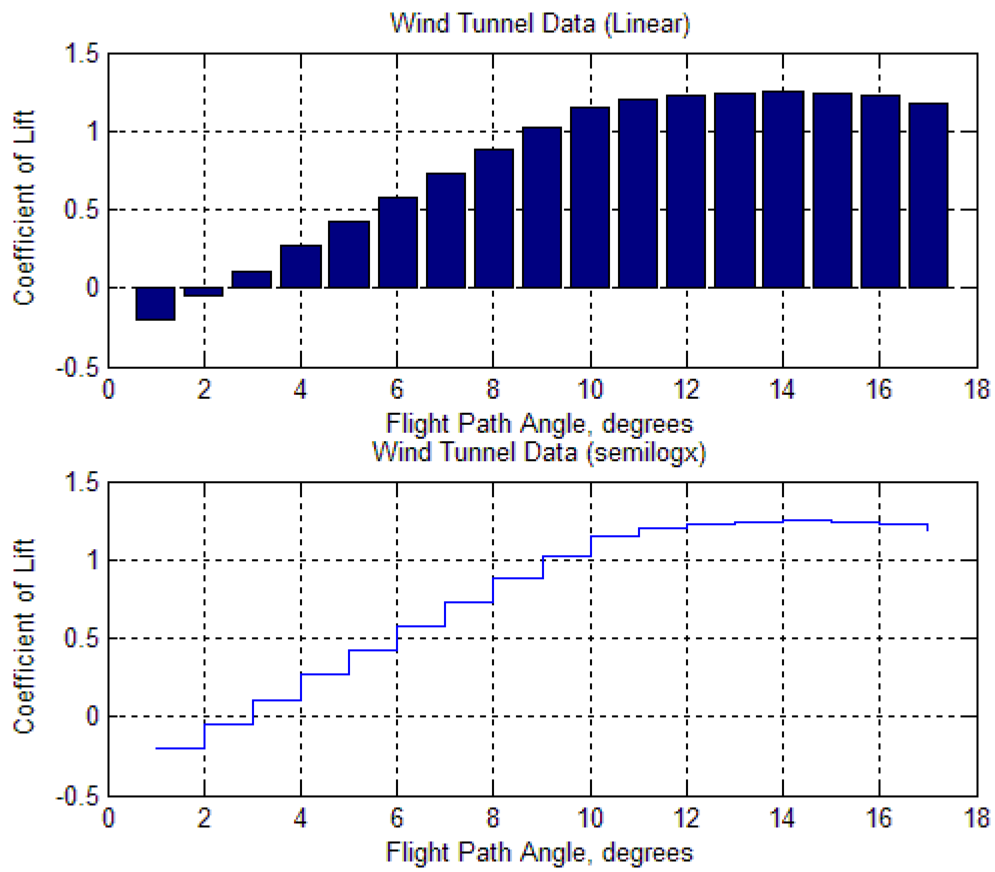
5. (π, 1)
6. (π/2, 0)
7. (2.3, 0.5)
8. (0.5, 0.5)

4.4 bar და stairs გრაფიკები

ნახ 7.5 წარმოადგენს **bar** და **stairs** გრაფიკებს, აგებულს აეროდინამიკური გვირაბის ექსპერიმენტის მონაცემებისათვის, რომელიც უკვე რამდენიმეჯერ შევხვდა განხილულ მაგალითებში.

bar(y)	ააგებს bar გრაფიკს y მნიშვნელობათათვის
bar(x,y)	ააგებს bar გრაფიკს y ვექტორის ელემენტებისათვის იმ მნიშვნელობებზე, რომელიც მოცემულია x ვექტორის სახით
stairs(y)	ააგებს stair გრაფიკს y ვექტორის მნიშვნელობათათვის.
stairs(x,y)	ააგებს stair გრაფიკს y ვექტორის ელემენტებისათვის იმ მნიშვნელობებზე, რომელიც მოცემულია x ვექტორის სახით

```
subplot(2,1,1), bar(y)
    title('Wind Tunnel Data (Linear)'),...
    xlabel('Flight Path Angle, degrees'),...
    ylabel('Coefficient of Lift'),...
    grid
    subplot(2,1,2),stairs(y)
    title('Wind Tunnel Data (semilogx)'),...
    xlabel('Flight Path Angle, degrees'),...
    ylabel('Coefficient of Lift'),...
    grid
```



ნახ 7.5. bar და stairs გრაფიკები

4.5 გრაფიკული ოფციები

MATLAB შეიცავს სხვადასხვა ოფციებს გრაფიკის გასაფორმებლად, გასაუმჯობესებლად. განვიხილოთ ზოგიერთი მათგანი, რომელიც შემდგონში ხშირად გამოგვადგება.

4.5.1 რამდენიმე მრუდის აგება ერთიდაიგივე ნახაზზე

არსებობს ერთიდაიგივე ნახაზზე რამდენიმე მრუდის აგების სამი გზა. ერთ-ერთი მათგანია: **plot** ბრძანების არგუმენტად ავიღოთ მატრიცები.

მეორე გზაა plot ბრძანებას მივცეთ რამდენიმე არგუმენტი, მაგალითად:

plot(x,y,w,z)

სადაც x, y, w, z ვექტორებია. როცა ამ ბრძანებას მივცემთ, MATLAB ააგებს ორივე გრაფიკს ერთ ნახაზზე (ერთ გრაფიკულ ფანჯარაში). ამ მეთოდის უპირატესობა ისაა, რომ საჭირო არაა წერტილების რაოდენობა ერთმანეთს ემთხვეოდეს. MATLAB ავტომატურად შეარჩევს მათთვის წირის განსხვავებულ ფორმას და ფერს.

მესამე გზა ერთიდაიგივე ნახაზზე სხვადასხვა მონაცემთა აგებისა ხორციელდება ბრძანებით **hold.** ეს ბრძანება ინარჩუნებს გრაფიკულ ფანჯარას აქტიურ მდგომარეობაში და ყოველი შემდგომი გრაფიკული ბრძანება ახალ გრაფიკს უმატებს მას, **hold** ბრძანების ხელმეორედ მიცემა შეწყვეტს ამ პროცესს.

4.5.2 წირის და აღნიშვნის სტილი

ბრძანება **plot(x,y)** აწარმოებს წირის აგებას, რომელიც აერთებს x და y ვექტორებით მოცემულ წერტილებს. შესაძლებელია აირჩიოთ და მიუთითოთ წირის ფორმა (წყვეტილი, წერტილოვანი და წყვეტილწერტილოვანი) და ფერი. შეგიძლიათ წირით სულაც არ შეაერთოთ წერტილები. ასთეთ შემთხვევაში მხოლოდ ვექტორებით განსაზღვრული წერტილები აღინიშნება ნახაზზე. შეგვიძლია მივუთითოთ აღნიშვნა წერტილის ადგილზე: ვარსკვლავი, წრე, სამკუთხედი და სხვა. შემდეგი ცხრილი შეიცავს წირისა და წერტილის აღნიშვნების მრავალფეროვან არჩევანს:

წირის ტიპი	განმსაზღვრელი	წერტილის ტიპი	განმსაზღვრელი
უწყვეტი	-	წერტილი	.
წყვეტილი	--	პლუსი	+
წერტილოვანი	:	ვარსკვლავი	*
წყვეტილ-წერტილოვანი	-.	წრე	O
		x - აღნიშვნა	x

საილუსტრაციოდ მოგვყავს მაგალითი. ბრძანება უწყვეტი წირით აერთებს წერტილებს, რომელთა კოორდინატები წარმოდგენილი x და y ვექტორების სახით და შემდეგ დასვამს ამ წირზე შესაბამის წერტილებს აღნიშვნით x:

```
plot(x, y, x, y, 'x')
```

გარდა ამისა, შეგიძლიათ შეარჩიოთ და მიუთითოთ წირის ფერი შემდეგი ცხრილის მიხედვით:

წირის ტიპი	განმსაზღვრელი
წითელი	**r**
მწვანე	**g**
ლურჯი	**b**
თეთრი	**w**
უხილავი	**i**

შემდეგი ბრძანება უწყვეტი ლურჯი წირით აერთებს წერტილებს, რომელთა კოორდინატები წარმოდგენილი x და y ვექტორების სახით და შემდეგ დასვამს ამ წირზე შესაბამის წერტილებს წითელი ფერის x ნიშნით:

```
plot(x, y, ‘b’, x, y, ‘xr’)
```

4.5.3 გრადაცია

MATLAB ავტომატურად ირჩევს ღერძების გრადაციას. მაგრამ შესაძლებელია მივუთითოთ ღერძების საზღვრები ბრძანებით **axis:**

axis	ეს ბრძანება გრაფიკულ ფანჯარაზე ‘გაყინავს’ ღერძების არსებულ გრადაციას, ავტომატურ რეჟიმში დასაბრუნებლად იგივე ბრძანებას ვიმეორებთ
axis(v)	v 4 ელემენტიანი ვექტორია, რომელიც შეიცავს გრადაციის სიდიდეებს [xmin, xmax, ymin, ymax]
axis(‘square’)	განსაზღვრავს ღერძების თანაფარდობას (კვადრატული)
axis(‘normal’)	განსაზღვრავს ღერძების თანაფარდობას (ნორმალური)

ეს ბრძანებები განსაკუთრებით სასარგებლოა სხვადასხვა გრადაციით(scaling) აგებული ნახაზების შესადარებლად.

4.5.4 ბრძანება subplot

ბრძანება **subplot** საშუალებას გვაძლევს გრაფიკული ფანჯარა დავყოთ რამდენიმე ქვეფანჯარად. ნახ 7.3 და ნახ 7.5 შექმნილია ამ ბრძანების გამოყენებით. **subplot** გააჩნია 3 არგუმენტი subplot (m n p), m,n განსაზღვრავს რამდენ ქვეფანჯრადაა დაყოფილი გრაფიკული ფაბჯარა და რა რიგით, ხოლო p მიუთითებს რომელ ფანჯარაში აიგოს მოცემული გრაფიკი. ფანჯრები დანომრილია მარცხნიდან მარჯვნივ, ზემოდან ქვემოთ. მაგალითად შემდეგი ბრძანებები განსაზღვრავს, რომ გრაფიკული ფანჯარა დაყოფილია ორ ქვეფანჯარად – ზედა და ქვედა, ხოლო მოცემული გრაფიკი აიგება ზედა ფანჯარაში:

```
subplot(211), plot(x,y)
```

4.6 ეკრანის კონტროლი

როგორც ვიცით MATLAB-ს ორი ძირითადი ფანჯარა აქვს ბრძანებათა ფანჯარა და გრაფიკული ფანჯარა შემდეგი ბრძანებები საშუალებას გვაძლევს შევარჩიოთ და გავასუფთაოთ ფანჯარა:

shg	ეკრანზე გამოიყვანს გრაფიკულ ფანჯარას
ნებისმიერი კლავიში	დააბრუნებს ეკრანზე ბრძანებათა ფამჯარას

clc	ასუფთავებს ბრძანებათა ფანჯარას
clf	ასუფთავებს გრაფიკულ ფანჯარას
home	ასუფთავებს ბრძანებათა ფანჯარის ხილულ ნაწილს ეკრანზე და კურსორი გადადის ბრძანებათა ფანჯარის ზედა მარცხენა კუთხეში

4.6.1 ბრძანება **ginput**

ეს ბრძანება საშუალებას გვაძლევს ავიღოთ კოორდინატები პირდაპირ გრაფიკული ფანჯარიდან მაუზის ან ისრიანი კლავიშების საშუალებით:

[x,y]=ginput	საშუალებას გვაძლევს შევარჩიოთ წერტილთა შეუზღუდავი რაოდენობა გრაფიკული ფანჯარიდან მაუზის ან კლავიშების საშუალებით. შეიქმნება x, y ვექტორები კოორდინატთა შესაბამისი მნიშვნელობებით. ბრძანების მოქმედებას წყვეტს კლავიში return key
[x,y]=ginput(n)	საშუალებას გვაძლევს შევარჩიოთ n წერტილი გრაფიკული ფანჯარიდან მაუზის ან კლავიშების საშუალებით. შეიქმნება x, y ვექტორები კოორდინატთა შესაბამისი მნიშვნელობებით. ბრძანების მოქმედებას წყვეტს კლავიში return key

4.6.2 გრაფიკის დაბეჭდვა

print	დაბეჭდავს მაღალი გარჩევის გრაფიკს პრინტერზე, ან შეინახავს მას როგორც .pdf ფაილს დისკზე.

პრობლემა - მეტეოროლოგიური რაკეტის ტრაექტორია

მეტეოროლოგიური რაკეტა გამოიყენება ატმოსფეროს სხვადასხვა ფენებში მიმდინარე პროცესების შესახებ მონაცემთა შესაგროვებლად. მაგალითად ოზონის შემცველობის განსაზღვრა. რაკეტაზე სხვა ხელსაწყოებთან ერთად დამაგრებულია ტელემეტრული სისტემა, როლის საშუალებითაც ხდება მონაცემების გადმოგზავნა დედამიწაზე. ყოველ ანათვლს თან ახლავს მონაცემები თავად რაკეტის შესახებ ამ მომენტისათვის: სიმაღლე, სიჩქარე და აჩქარება.

დავუშვათ გვაქვს მონაცემთა ფაილი, რომელიც შეიცავს ინფორმაციას იონოსფეროს გამოსაკვლევედ გაშვებული ორსაფეხურიანი (two-stage) მეტეოროლოგიური რაკეტის სიმაღლის, სიჩქარის და აჩქარების შესახებ. ვიცით, რომ რაკეტის პირველი საფეხურის საწვავი დაიწვა 35 წამში, რის შემდეგაც რაკეტის სიჩქარემ მიაღწია 1 250 მეტრს წამში. ამის შემდეგ 2 წუთის განმავლობაში რაკეტა განიცდის თავისუფალ ვარდნას და აღწევს იონოსფეროს დაბალ ფენებს 100 კილომეტრის სიმაღლეზე. ამ დროისათვის გრავიტაცია შეანელებს რაკეტის სიჩქარეს 100 მეტრ/წამამდე. ამის შემდეგ ირთვება მეორე საფეხურის საწვავის მექანიზმი, რომელიც გამოიწვევს რაკეტის აჩქარებას. იგი გაიჭრება იონოსფეროს მაღალ ფენებში. უნდა ავაგოთ მრუდი ფაილში ჩაწერილი მონაცემების მიხედვით, რათა შევადაროთ ისინი თერიულად გათვლილ ტრაექტორიას.

1. **ამოცანის დასმა**

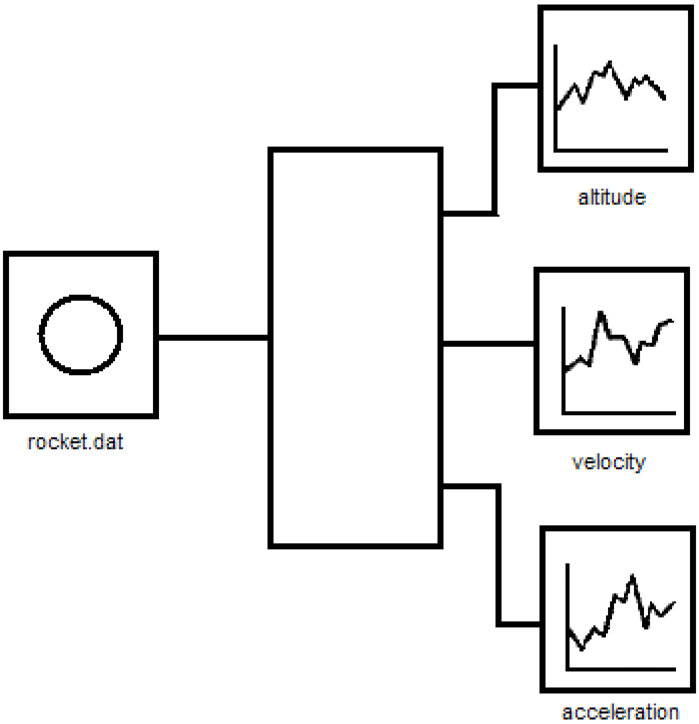
ავაგოთ რაკეტის სიმაღლის, სიჩქარის, და აჩქარების მრუდი, ფაილიდან წაკითხულ მონაცემთა მიხედვით.

2. INPUT/OUTPUT აღწერა

ცხრილი შეიცავს მონაცემებს რაკეტის სიმაღლის, სიჩქარის და აჩქარების მნიშვნელობბებს დროის შესაბამისი მომენტუსათვის.

დრო [წმ]	სიმაღლე [მ*10^5]	სიჩქარე [მ/წმ]	აჩქარება[მ/წმ^2]
10	0	500	42
20	0.1100	895	40
30	0.2500	1245	10
40	0.3400	1250	-9
50	0.4200	1147	-10
60	0.5300	1046	-10
70	0.6200	920	-10
80	0.6900	824	-10
90	0.7500	711	-10
100	0.8300	609	-10
110	0.9200	501	-10
120	0.9900	400	-10
130	1.0300	286	-10
140	1.0700	190	-10
150	1.1300	90	-5
160	1.1500	264	30
170	1.2100	741	56
180	1.3000	1381	78
190	1.3700	2141	94
200	1.6100	3195	118
210	2.0000	3791	19
220	2.3500	3811	-10
230	2.7600	3698	-10
240	3.3000	3560	-10

ნახ 7.6 ნახაზზე წარმოდგენილია განსახილველი ამოცანის INPUT/OUTPUT დიაგრამა, რომელიც გვიჩვენებს, რომ საწყისი მონაცემები ფაილის სახითაა მოცემული, ხოლო შედაგად უნდა მივიღოთ მონაცემთა მიხედვით აგებული გრაფიკები.



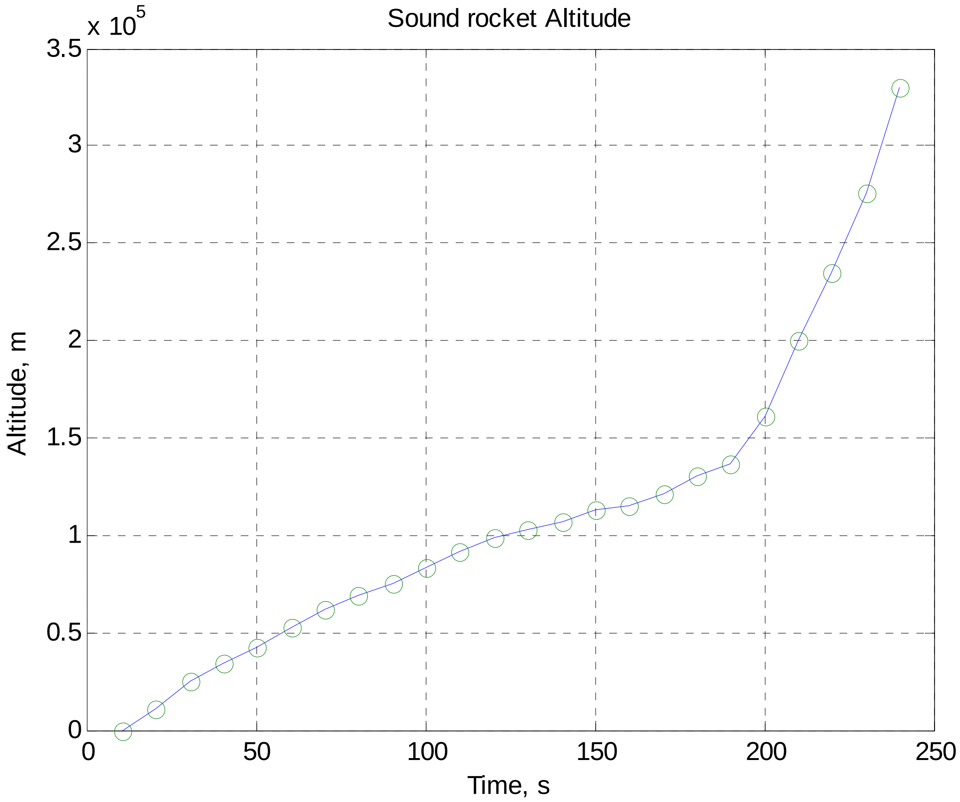
ნახ 7.6. I/O დიაგრამა

3. სახელდახელო ამოხსნა

რადგან ამ ამოცანაში არაფერს არ ვითვლით, გამოსათვლელიც არაფერია. მაგრამ შეგვიძლია ამოვიღოთ ფაილიდან რამდენიმე მონაცემი და მიღებული შედეგის მიხედვით გავაკეთოთ დასკვნა.

4. MATLAB ამოხსნა

MATLAB მრავალფეროვანი გრაფიკული შესაძლებლობების გამო ასეთი გრაფიკების აგება მარტივად ხდება. ავაგოთ წირი და შემდეგ დავსვათ წერტილთა შესაბამისი აღნიშვნები.

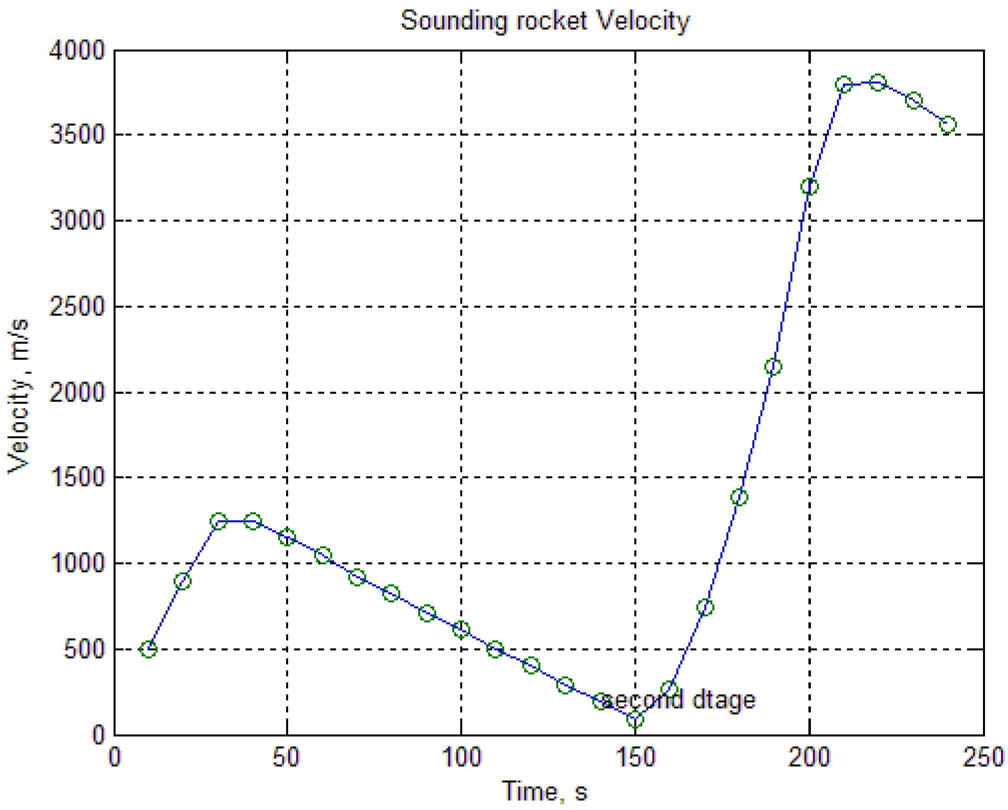


ნახ 7.7. მეტეოროლოგიური რაკეტის სიმაღლე

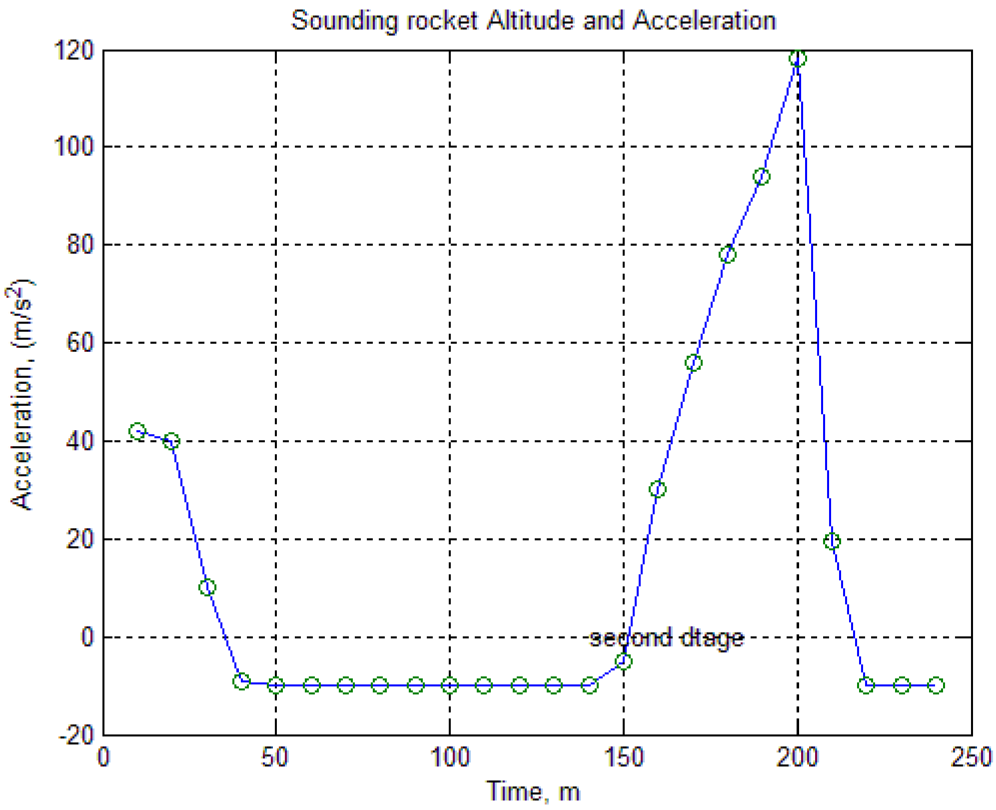
```
%
%     This program generates plots of the altitude,
%     velocity, and acceleration of sounding rocket
%
clear; clc;
load rocket.dat
time=rocket(:,1);
alt=rocket(:,2);
vel=rocket(:,3);
acc=rocket(:,4);
%
%     These commands generate and label
%     a plot of the altitude data
%

plot(time,alt,time,alt,'o'),...
title('Sound rocket Altitude'),...
xlabel('Time, s'),...
ylabel('Altitude, m'),...
grid
pause
%
%     These commands generate and label
%     a plot of the velocity data
```

```
%
plot(time,vel,time,vel,'o'),...
title('Sounding rocket Velocity'),...
xlabel('Time, s'),...
ylabel('Velocity, m/s')
text(140,200,'second dtage')
grid
pause
%
%     These commands generate and label
%     a plot of the acceleration data
%
plot (time,acc,time,acc,'o')
title('Sounding rocket Altitude and Acceleration'),...
xlabel('Time, m'),...
ylabel('Acceleration, (m/s^2)')
text(140,0,'second dtage')
grid
```



ნახ 7.8. მეტეოროლოგიური რაკეტის სიჩქარე



ნახ 7.9. მეტეოროლოგიური რაკეტის აჩქარება

5. შემოწმება

მონაცემები ჩაწერილია ფაილში **rocket.dat**. 7.7 – 7.9 ნახაზზე წარმოდგენილია შედეგად მიღებული გრაფიკები მეტეოროლოგიური რაკეტის სიმაღლის, სიჩქარისა და აჩქარებისათვის შესაბამისად. სიჩქარის გრაფიკიდან კარგად ჩანს, რომ თავდაპირველად იგი იზრდება, პირველი საფეხურის საწვავის დაწვის შემდეგ – თანდათან კლებულობს, ხოლო როცა მეორე საფეხურის საწვავი იწყებს წვას, ისევ მომატებას იწყებს. აჩქარების გრაფიკიდან ადვილად შეამჩნევთ აჩქარებას, რომელიც გამოწვეულია პირველი საფეხურის და მეორე საფეხურის გამო. საწვავის ამოწურვის სტადიაში ორივე შემთხვევაში რაკეტის აჩქარება –9.5 მ/წმ-ია, რაც გრავიტაციით გამოწვეული თავისუფალი ვარდნის აჩქარებაა. ნახ. 7.8 და 7.9 **text** ბრძანების საშუალებით მივუთითეთ მეორე საფეხურის დასაწყისის შესაბამისი წერტილი გრაფიკზე.

4.7 სამგანზომილებიანი გრაფიკი

MATLAB –ში სამგანზომილებიანი გრაფიკის სხვადასხვა გზა არსებობს. შესაძლებელია ავაგოთ ბადურა (mesh) ზედაპირი სამგანზომილებიან სივრცეში. შეგვიძლია ასეთ ზედაპირს შევხედოთ გრაფიკულად ნებისმიერი მიმართულებიდან, შეგვიძლია განვსაზღვროთ ღერძების გრადაცია. სამგანზომილებიანი ზედაპირი შეიძლება ავაგოთ როგორც ზედაპირის კონტური სხვადასხვა ჭრილში. სამგანზონმილებიანი გრაფიკის ყველა ფორმა მნიშვნელოვან ინფორმაციას იძლევა სამგანზომილებიან მონაცემთა ანალიზისათვის.

4.7.1 ბადურა ზედაპირი (mesh ზედაპირი)

mesh ზედაპირი აიგება მატრიცის სახით წარმოდგენილი მონაცემების საფუძველზე. მატრიცის თითოეული ელემენტი შესაბამება წერტილს ბადეზე.

იმისათვის, რომ შევქმნათ მონაცემები სამგანზომილებიანი ზედაპირისათვის, პირველ რიგში განვსაზღვრავთ დამოუკიდებელი ცვლადების x და y სიდიდეების მწკრივს, შემდეგ გამოვითვლით z სიდიდეებს, რომელიც x და y –ს ფუნქციაა და წარმოადგენს სამგანზომილებიან ზედაპირს. x და y ისე ავირჩევთ, რომ მათი მნიშვნელობები თანაბრად იყოს განაწილებული x-y სიბრტყეზე. მაგალითად, დავუშვათ გვინდა ავაგოთ ფუნქცია

$$f(x,y) = z = \left|\sqrt{1 - x^2 - y^2}\right|$$

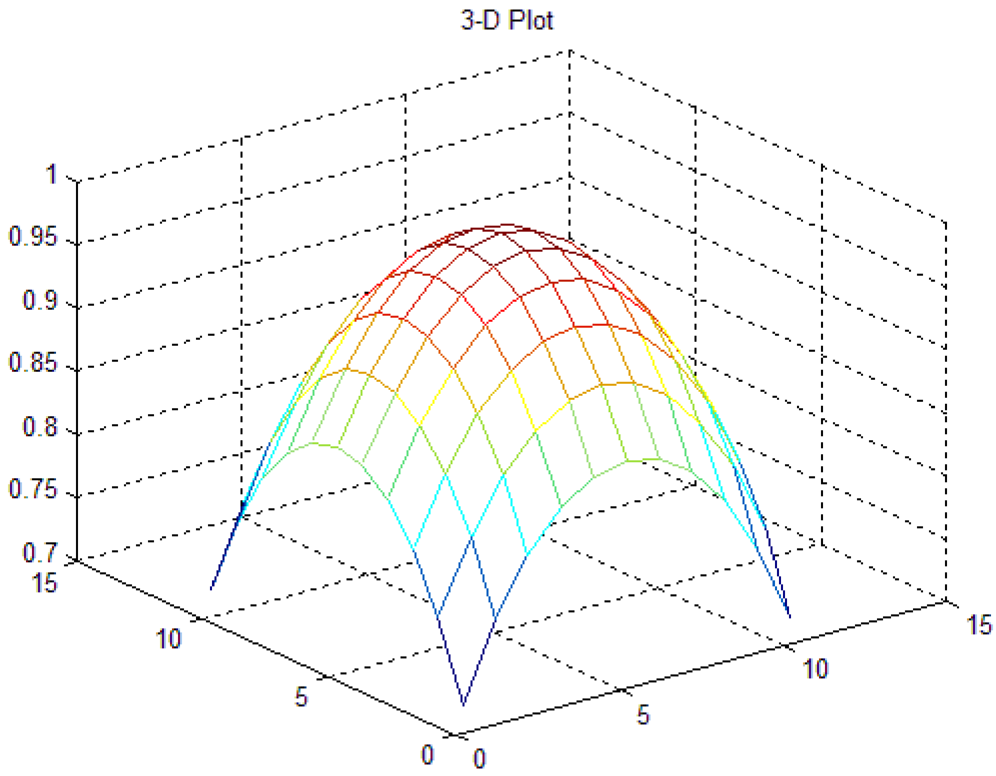
ისე, რომ $-0.5 \le x \le 0.5$ და $-0.5 \le y \le 0.5$

ეს ფუნქცია წარმოადგენს ერთეულოვანი რადიუსის მქონე სფეროს ზედაპირის განტოლების სახეცვლილებას:

$$x^2 + y^2 + z^2 = 1$$

რადგან *f(x,y)* იყენებს კვადრატული ფესვის მხოლოდ დადებით მნიშვნელობებს, იგი ამ ფუნქციის მხოლოდ ზედა ნახევარს წარმოადგენს. იმისათვის, რომ ავაგოთ სამგანზომილებიანი ზედაპირი შემდეგნაირად უნდა მოვიქცეთ:

```
for m =  1:11
    x = (m-6)*0.1;
    for n = 1:11
        y = (n-6)*0.1;
        z(m,n) = sqrt(abs(1 - x.^2 - y.^2));
    end
end
mesh(z),...
title('3-D Plot')
grid
```



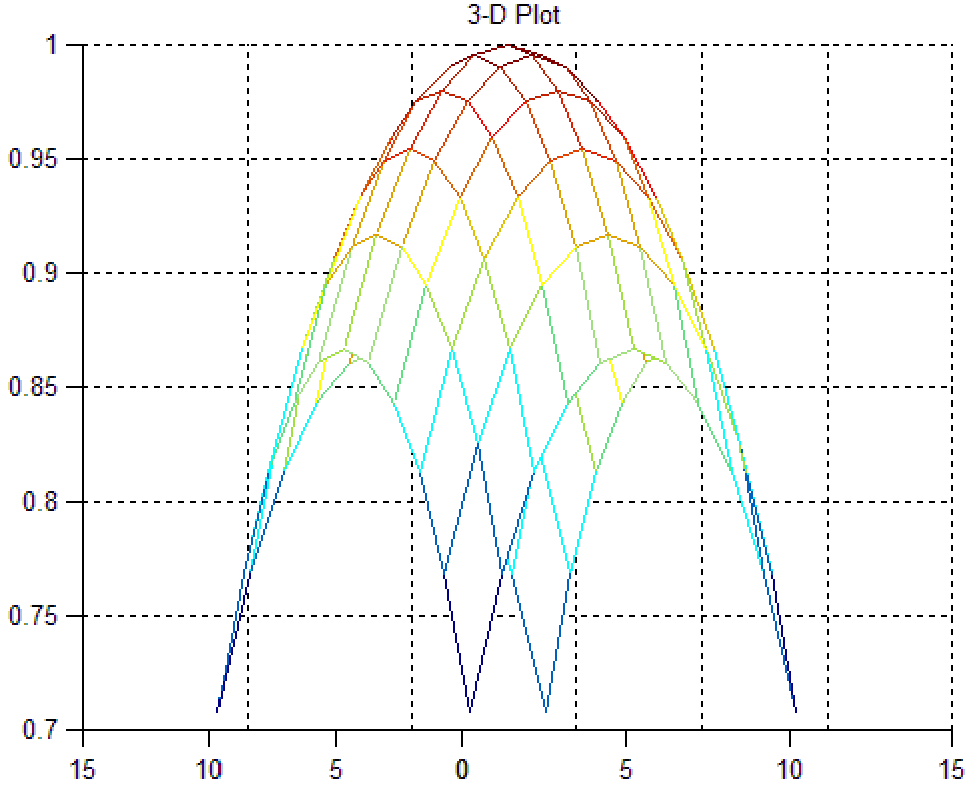
ნახ 7.10. სფეროს ნაწილის სამგანზომილებიანი გრაფიკი

z ვექტორის აგების მეორე გზაც არსებობს. ვსარგებლობთ ბრძანებით meshgrid, რომლის არგუმენტები x და y ვექტორის მნიშვნელობებია. იგი აწარმოებს 2 მასივს x და y მნიშვნელობების მიხედვით. სფეროს ზემოთაღწერილი ნაწილი ასეც შეგვიძლია ავაგოთ:

```
[X,Y]= meshgrid(-0.5:0.1:0.5,-0.5:0.1:0.5);
Z=sqrt(abs(1-X.^2-Y.^2));
mesh(Z)
grid
```

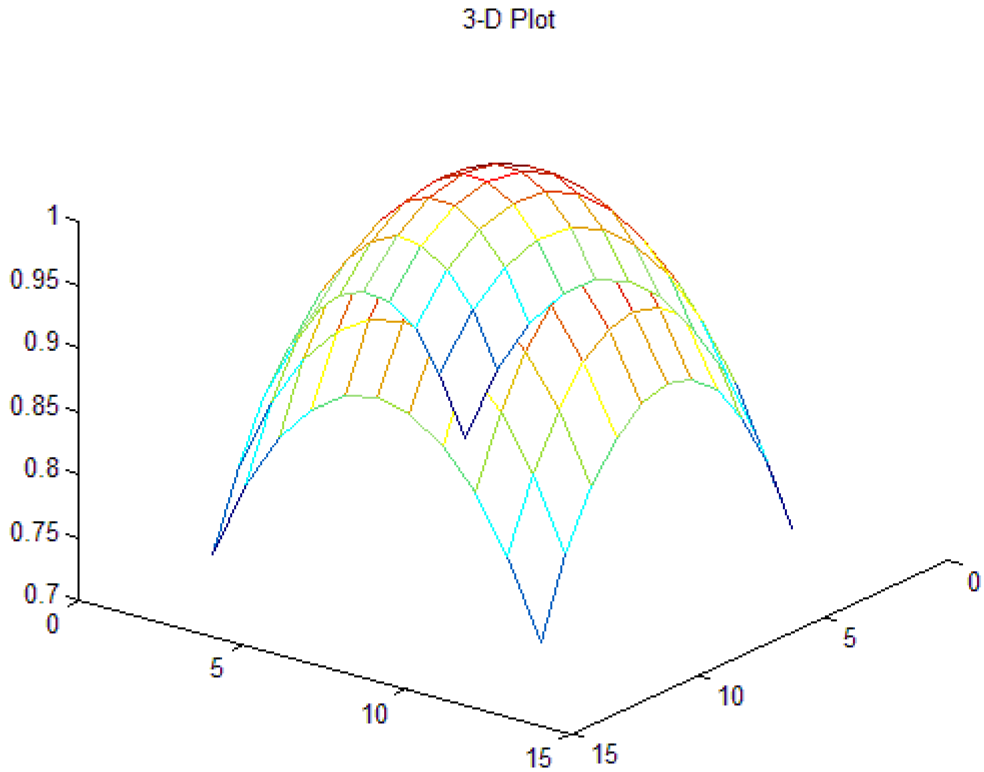
შეიძლება სამგანზომილებიანი გრაფიკის აგებისას საჭირო იყოს აგებულ ზედაპირს რაიმე გარკვეული მიმართულებიდან შევხედოთ. ხედვის მიმართულება განისაზღვრება გრადუსებში გამოხატული აზიმუტისა (ჰორიზონტალური) და სიმაღლის(ვერტიკალური) მიხედვით. აზიმუტი = 0, სიმაღლე = 0 შეესაბამება მატრიცის ქვედა მარჯვენა კუთხე. აზიმუტის დადებითი მიმართულება საათის ისრის საწინააღმდეგო მიმართულებაა, z ღერძის გასწვრივ. დადებითი სიმაღლეები გვიჩვენებს ზედაპირს ზემოდან, უარყოფითი – ქვემოდან.

თუ არ ვუთითებთ ხედვის კუთხეს მაშინ მას აქვს (default) მნიშვნელობა აზიმუტი –37.5 გრადუსი და სიმაღლე 30 გრადუსი ნახ 7.10. ხედვის კუთხე განისაზღვრება როგორც **mesh** ბრძანების მეორე არგუმენტი mesh(x, [–37.5, 0]). მაგალითად, ნახ 7.11 ხედვის კუთხეა [–37.5, 0]



ნახ 7.11. სფეროს ნაწილი. ხედვის კუთხე (-37.5, 0)

ნახ 7.12 ნახაზზე გრაფიკი ნაჩვენებია მიმართულებიდან: [-37.5, -30)



ნახ 7.12. სფეროს ნაწილი 30 გრადუსიანი სიმაღლიდან

სავარჯიშო

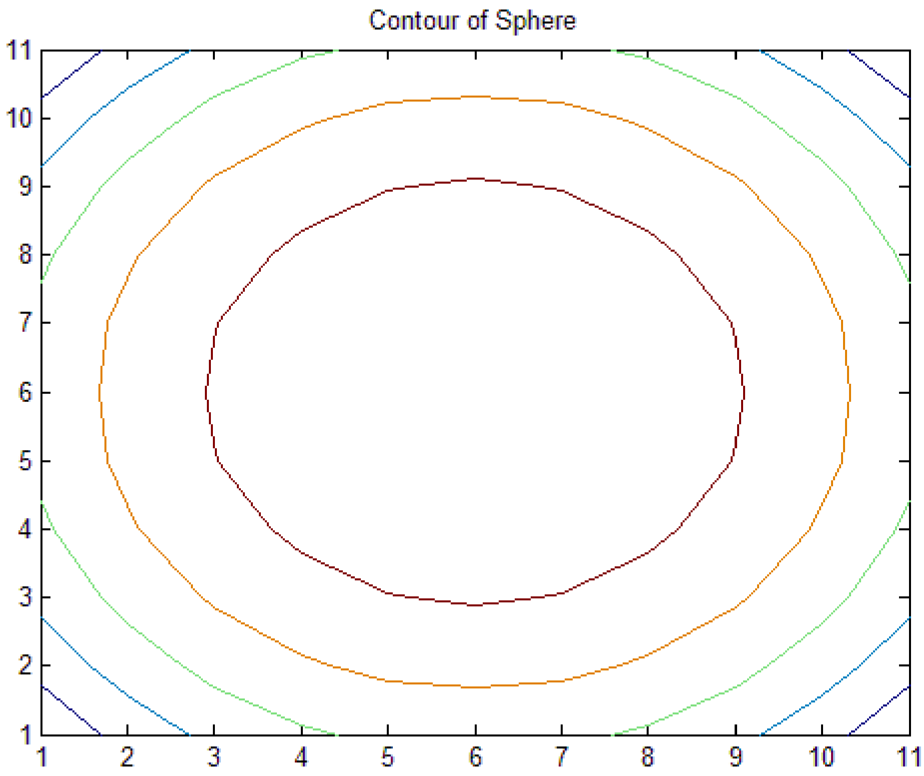
ააგე **mesh** ზედაპირი, რომელიც განისაზღვრება შემდეგი ფორმულით. ვიდრე ააგებდეთ. შეეცადეთ წარმოიდგინოთ, როგორი იქნება იგი.

1. $f(x,y) = \left|\sqrt{1-x^2-y^2}\right| \quad for \quad 0 \le x \le 0.5 \quad and \quad for \quad 0 \le y \le 0.5$
2. $f(x,y) = -\left|\sqrt{1-x^2-y^2}\right| \quad for \quad -0.5 \le x \le 0.5 \quad and \quad for \quad -0.5 \le y \le 0.5$
3. $f(x,y) = \left|\sqrt{1-x^2-y^2}\right| \quad for \quad -0.5 \le x \le 0 \quad and \quad for \quad 0 \le y \le 0.5$
4. $f(x,y) = \left|\sqrt{1-x^2-y^2}\right| \quad for \quad -1 \le x \le 1 \quad and \quad for \quad -1 \le y \le 1$

4.7.2 კონტურული გრაფიკი

სიმაღლეების (elevation) რუქა შეიცავს წირების ჯგუფს, რომლებიც აერთებენ ერთნაირ სიმაღლეზე მყოფ წერტილებს. თუ დედამიწის ფიზიკურ რუქაზე ასეთი წირებით შევაერთებთ ზღვის დონიდან ერთი სიმაღლის მქონე წერტილებს, შეგვიძლია ვიმსჯელოთ მათი რეალური სიმაღლის შესახებ. ასეთი ტიპის რუქებს კონტურულს უწოდებენ. MATLAB-ის საშუალებით მატრიცის სახით მოცემული ზედაპირისათვის შეგვიძლია ავაგოთ მსგავსი კონტურული გამოსახულება. ამისათვის არსებობს ბრძანება **contour**:

contour(z)	აგებს z მატრიცით მოცემული ზედაპირის კონტურს. კონტურული წირების რაოდენობა და მათი მნიშვნელობები ავტომატრად შეირჩევა MATLAB –ის მიერ. ნახაზის ზედა მარცხენა კუთხე შეესაბამება სიდიდეს მდებარეობაში z(1,1)
contour(z,n)	აგებს z მატრიცით განსაზღვრული ზედაპირის n დონის კონტურს
contour(z,v)	აგებს z მატრიცით განსაზღვრული ზედაპირის კონტურულ გამოსახულებას კონტურული წირებით, რომელთა დონეები მოცემულია v ვექტორის სახით

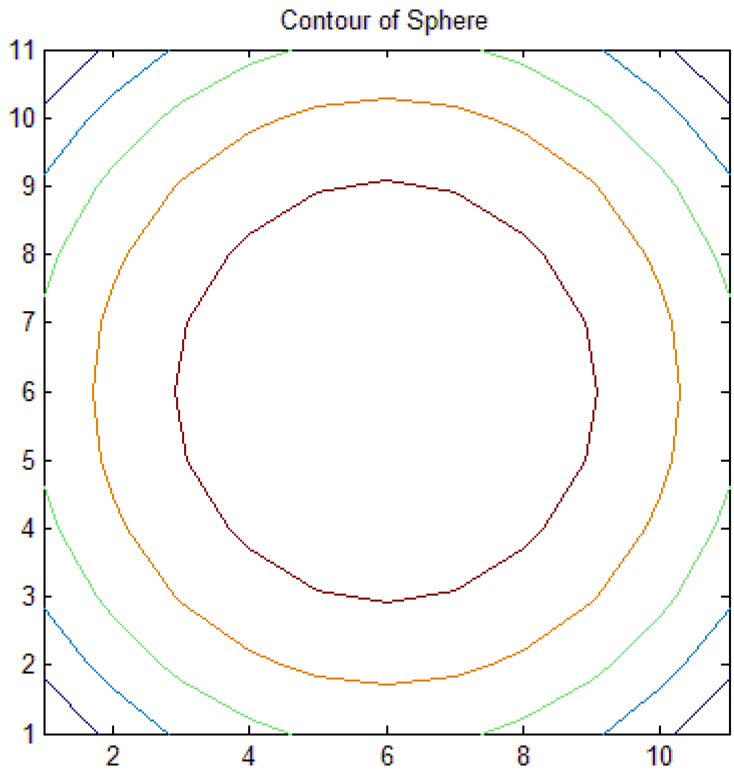


ნახ 7.13. გრაფიკი ზედაპირის 5 სხვადასხვა დონის კონტურით

ამ ბრძანებას შესაძლოა ჰქონდეს არგუმენტები, რომლებიც განსაზღვრავს კონტურთა დონეს და ღერძების მასშტაბირებას. შემდეგი ბრძანებები აგებს სფერული ზედაპირის 7.13 გრაფიკს:

```
[X,Y]= meshgrid(-0.5:0.1:0.5,-0.5:0.1:0.5);
Z=sqrt(abs(1-X.^2-Y.^2));
contour(Z,5),...
title('Contour of Sphere')
```

თუ ამ ბრძანებებს გავუშვებთ axis(‘square’) ბრძანების შემდეგ მივიღებთ ნახ 7.14.



ნახ 7.14. სფეროს ზედაპირის კონტური ღერძების კვადრატული თანაფარდობით

პრობლემა: **terrain navigation** ტოპოგრაფიული გაზომვები?????

terrain navigation ძირითადი კომპონენტია დისტანციურად მართვადი ობიექტების წარმოებაში. ასეთებია მაგალითად რობოტი, უპილოტო თვითმფრინავი და რაკეტა, თვითმართვადი წყალქვეშა ხომალდი და სხვა. ამ მოწყობილობებს ბორტზე გააჩნიათ კომპიუტერი, რომელშიც შეტანილია ინფორმაცია იმ გარემოს შესახებ, სადაც ობიექტი მოძრაობს. ვიცით რა დროის ნებისმიერ მომენტში ობიექტის მდებარეობა შესაძლებელია შეირჩეს დანიშნულების ადგილამდე მისასვლელი ოპტიმალური გზა. როცა დანიშნულების ადგილი შეიცვლება, კომპიუტერი მიმართავს ინფორმაციას გარემოს შესახებ და გამოითვლის ახალ მიმართულებას.

კომპიუტერული პროგრამა, რომელიც ასეთ სისტემებს მართავს, წინასწარ გამოიცდება სხვადასხვაგვარ ტოპოგრაფიულ პირობებში. მონაცემთა ბაზებში არსებობს დედამიწისა თუ ოკეანის ფსკერის შესახებ ტოპოგრაფიული ინფორმაცია. იმისათვის რომ განისაზღვროს გასავლელი გზის სირთულე საჭიროა განისაზღვროს პიკების რაოდენობა. პიკი ეს არის წერტილი, რომელის სიმაღლეც ყველაზე მეტია მის ირგვლივ წერტილების მიმართ.

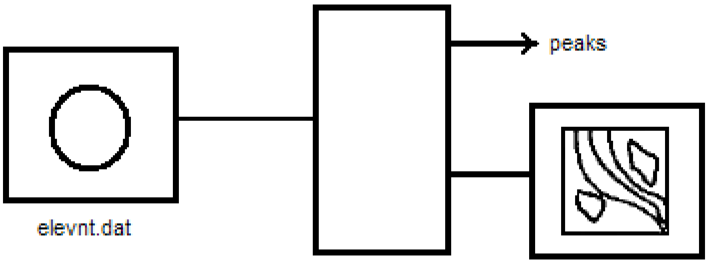
დაწერეთ პროგრამა, რომელიც წაიკითხავს ტოპოგრაფიულ მონაცემებს ფაილიდან და დაბეჭდავს პიკების მდებარეობას საკოორდინატო ბადეზე. ასევე დაბეჭდავს სამგანზომილებიან mesh და კონტურულ გრაფიკს ფაილში არსებული მონაცემების საფუძველზე.

1. ამოცანის დასმა

დგანსაზღვრეთ პიკების რაოდენობა და მდებარეობა ფალში ჩაწერილი მონაცემების საფუძველზე.

2. INPUT/OUTPUT აღწერა

ნახ 7.17 შეიცავს INPUT/OUTPUT დიაგრამას. დიაგრამიდან ჩანს, რომ საწყისი მონაცემები წარმოდგენილია ფაილის სახით, ხოლო შედეგი – პიკების მდებარეობა ცხრილის სახით და მონაცემთა საფუძველზე აგებული კონტურული გრაფიკი.



ნახ 7.15. I/O დიაგრამა

3. სახელდახელო ამოხსნა

დავუშვათ ზედაპირი დაყოფილია 6 × 8 კვადრატად და გაზომილია თითოეული მათგანის სიმაღლე. მონაცემები ჩაწერილია მატრიცის სახით:

25	59	63	23	21	34	21	50
32	45	43	30	<u>37</u>	32	30	27
34	38	38	39	36	28	28	35
40	<u>45</u>	42	<u>48</u>	32	30	27	25
39	39	40	42	48	<u>49</u>	25	30
31	31	31	32	32	33	44	35

პიკის წერტილების შესაბამისი მნიშვნელობები ხაზგასმულია, ისე, რომ პიკების მდებარეობები შეესაბამება მატრიცის ელემენტებს (2,5), (4,2), (4,4), და (5,6).

4. MATLAB ამოხსნა

პიკების ძიება გულისხმობს ბადის მხოლოდ შიდა წერტილებს. კიდის წერტილი არ შეიძლება ჩაითვალოს პიკად იმიტომ, რომ არ ვიცით მისი ყველა მეზობელი წერტილის მნიშვნელობა. განვიხილოთ სიმაღლე წერტილში, რომელიც შეესაბამება მონაცემს elevation (m,n). მისი მოსაზღვრე მდებარეობები ნაჩვენებია ნახ 7.16 ამდენად, elevant(m,n) პიკი იქნება იმ შემთხვევაში, თუ თუ სრულდება პირობაები:

```
elevation(m,n) > elevation(m,n-1)
elevation(m,n) > elevation(m-1,n)
elevation(m,n) > elevation(m,n+1)
```

```
elevation(m,n) > elevation(m+1,n)
```

	elevation (m-1,n)	
elevation(m,n-1)	elevation (m,n)	elevation(m,n+1)
	elevation (m+1,n)	

ნახ 7.16. elevtion(m,n) მეზობელი წერტილების მდებარეობები

პროგრამის დაწერისას ვიყენებთ ერთმანეთში ჩალაგებული for ციკლის ოპერატორს.

```
%     This program reads the elevation data for
%     a land grid and determines the peaks.
%     It also ganarates a contour plot
%
load elevant.dat
elevation = elevant;
%
%     Identify rhe peaks.
%
[rows, cols] = size(elevation);
for m=2:rows-1
    for n=2:cols-1
        if (elevation(m,n)>elevation(m,n-1)&...
           elevation(m,n)>elevation(m-1,n)&...
           elevation(m,n)>elevation(m,n+1)&...
           elevation(m,n)>elevation(m+1,n))
                fprintf('Peak at (%6.0f, %6.0f) \n',m,n)
         end
    end
end
%
%     Generate plot
%
contour(elevation),...
title('Elevation Data')
```

5. შემოწმება

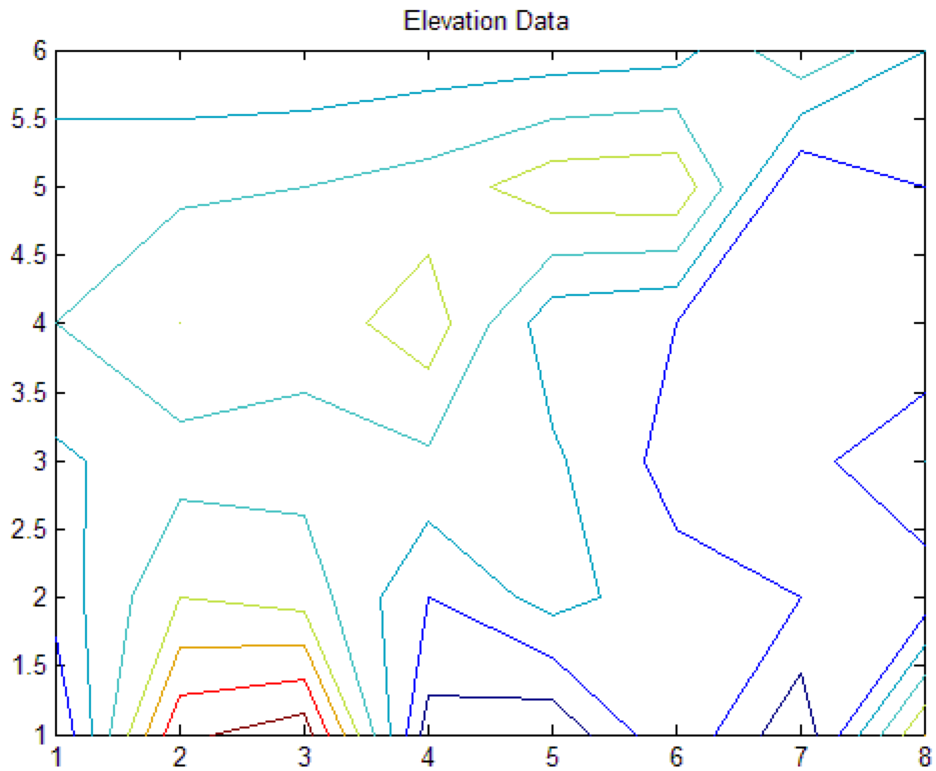
შევამოწმოთ პროგრამა ზემოთ განხილული მონაცემებისათვის, რომელიც ჩაწერილია ფაილში elevant.dat

პიკებისათვის მივიღებთ:

```
Peak at (     2,      5)
```

```
Peak at (      4,       2)
Peak at (      4,       4)
Peak at (      5,       6)
```

როგორც მოსალოდნელი იყო, ხოლო შესაბამისი გრაფიკი მოცემულია ნახ 7.17-ზე.



ნახ 7.17. ტოპოგრაფიულ მონაცემთა კონტურული გრაფიკი

ამ თავში მიმოვიხილეთ MATLAB გრაფიკული ბრძანებები. x-y გრაფიკი, რომელიც ყველაზე ხშირად გამოიყენება. MATLAB საშუალებას იძლევა გამოვიყენოთ მისი როგორც წრფივი, ასევე ლოგარითმული ფორმა. ასევე სასარგებლო და საჭიროა ერთიდაიმავე გრაფიკულ ფანჯარაში რამდენიმე გრაფიკის აგება მათი ერთმანეთთან შედარების თვალსაზრისით.

მრავალ ამოცანაში ფართოდ იყენებენ სამგანზომილებიან მონაცემებს, რომლებიც შეგვიძლია ავაგოთ MATLAB-ში როგორც სამგანზომილებიანი ზედაპირი.

4.8 ბრძანებები და ფიუნქციები

axis	აკონტროლებს ღერძების მასშტაბს
bar	აგებს bar გრაფიკს
clc	ასუფთავებს ბრძანებათა ფანჯარას

clf	ასუფთავებს გრაფიკულ ფანჯარას
contour	აგებს კონტურულ გრაფიკს
ginput	იღებს კოორდინატებს გრაფიკული ფანჯრიდან
grid	გრაფიკს უმატებს საკოორდინატო ბადეს
gtext	საშუალებას გვაძლევს ჩავწეროთ ტექსტი უშუალოდ გრაფიკულ ფანჯარაში
hold	ტოვებს მიმდინარე გრაფიკს გრაფიკულ ფანჯარაში
home	ბრძანებების ფანჯარაში კურსორი გადააქვს home მდებარეობაში
loglog	აგებს log-log გრაფიკს
mesh	ააგებს სამგანზომილებიან mesh გრაფიკს
meshgrid	აწარმოებს ვექტორებს საკოორდინატო ბადიათვის
plot	აგებს წრფივ x-y გრაფიკს
polar	აგებს პოლარულ გრაფიკს
print	ბეჭდავს მაღალი გარჩევის გრაფიკს
semilogx	აგებს ლოგარითმულ-წრფივ გრაფიკს
semilogy	აგებს წრფივ-ლოგარითმულ გრაიკს
shg	ეკრანზე გამოყავს გრაფიკული ფანჯარა
stairs	აგებს საფეხუროვან (stair) გრაფიკს
subplot	დაყოფა გრაფიკულ ფანჯარას რამდენიმე ქვეფანჯარად
text	ჩაწერს მითითებულ ტექსტს გრაფიკულ გამონასახში
title	გრაფიკს თავზე წააწერს სათაურს
xlabel	დააწერს მითითებულ აღნიშვნას x ღერძს
ylabel	დააწერს მითითებულ აღნიშვნას y ღერძს
surf	აგებს ზედაპირის ფერად, პარამეტრულ გამოსახულებას
surfc	იგივე surf მაგრამ უმატებს გრაფიკს შესაბამის კონტურულ გამოსახულებას
shading	აკონტროლებს ზედაპირზე ფერების სიგლუვეს(shading)
colormap	განსაზღვრავს ფერთა გამს (დიაპაზონს)

პრობლემები

1-13 პრობლემები დაკავშირებულია ამ თავში განხილულ ამოცანებთან, ხოლო 14 – 22 სხვა ამოცანებს უკავშირდება.

მეტეოროლოგიური რაკეტის ტრაექტორია. ეს ამოცანები დაკავშირებულია ამ თავში განხილულ პრობლემასთან მეტეოროლოგიური რაკეტის (ზონდის) შესახებ.

1. ააგეთ სიჩქარის წრფივ – ლოგარითმული გრაფიკი. შეადარეთ იგი იგივე მონაცემების წრფივ გრაფიკს
2. ააგეთ აჩქარების წრფივ – ლოგარითმული გრაფიკი. შეადარეთ იგი იგივე მონაცემების წრფივ გრაფიკს.
3. ააგეთ სიმაღლისა და სიჩქარის გრაფიკი ერთიდაიგივე ღერძების მიმართ, ერთ ნახაზზე, ხედავთ რაიმე ინფორმაციას, რომელიც თვალსაჩინო არ იყო მათი ცალკე ნახაზზე აგების შემთხვევაში?

4. ააგეთ სიმაღლისა და აჩქარების გრაფიკი ერთიდაიგივე ღერძების მიმართ, ერთ ნახაზზე, ხედავთ რაიმე ინფორმაციას, რომელიც თვალსაჩინო არ იყო მათი ცალკე ნახაზზე აგების შემთხვევაში?
5. ააგეთ სიჩქარის და აჩქარების გრაფიკი ერთიდაიგივე ღერძების მიმართ, ერთ ნახაზზე, ხედავთ რაიმე ინფორმაციას, რომელიც თვალსაჩინო არ იყო მათი ცალკე ნახაზზე აგების შემთხვევაში?
6. წააწერეთ ტექსტი რიმელიც უჩვენებს პირველი ეტაპის საწვავის ამოწურვის მომენტს სიჩქარისა და აჩქარების გრაფიკზე.
7. აჩქარების გრაფიკზე აღნიშნეთ ის ინტერვალები, როცა იგი გამოწვეულია მხოლოდ გრავიტაციით.

ტოპოგრაფიული (რელიეფური) ნავიგაცია

8. შეცვალე ამ პრობლემასთან დაკვშირებით შედგენილი პროგრამა ისე, რომ რომ დაბეჭდოს პიკების რაოდენობა რელიეფურ ბადეზე (elevation grid)
9. შეცვალე პროგრამა ისე, რომ მან დაბეჭდოს პიკების ნაცვლად (ღრმულების) მეზობელი წერტილების მიმართ ყველაზე ნაკლები სიმაღლის მქონე წერტილების მდებარეობა ბადეზე
10. შეცვალე პროგრამა ისე, რომ მოგვცეს ბადეზე უმაღლესი და უდაბლესი წერილის მდებარეობა და მნიშვნელობა.
11. შეცვალე პროგრამა ისე, რომ მოგვცეს სიმაღლეთა ბადის საშუალო სიმაღლე.
12. დავუშვათ წერტილებს შორის მანძილი ვერტიკალური და ჰორიზონტალური მიმართულებით 100 ფუტია, იპოვეთ ფუტებში გამოსახული ყოველი პიკის მდებარეობა ბადის ზედა მარცხენა კუთხის მიმართ.
13. დავუშვათ წერტილებს შორის მანძილი ვერტიკალური და ჰორიზონტალური მიმართულებით 100 ფუტია, იპოვეთ ფუტებში გამოსახული ყოველი პიკის მდებარეობა ბადის ქვედა მარცხენა კუთხის მიმართ.

ლოგარითმული გრაფიკი. მრავალ გამოყენებით ამოცანაში გვჭირდება ექსპერიმენტული ანათვლების აღება, რომელიც შემდეგ მათემეტიკურად უნდა აღვწეროთ, განვსაზღვროთ განტოლება, რომელიც მიღებულ მონაცემებს ასახავს, ასე ვთქვათ შევქმნათ პროცესის მათემატიკური მოდელი. მაგალითად, თუ ავაგეთ x-y გრაფიკი და ჩანს, რომ იგი ახლოსაა წრფესთან, შეგვიძლია შევაფასოთ მისი დახრის კუთხე და ღერძებთან გადაკვეთის წერტილები და ამის მიხედვით დავწეროთ წრფის განტოლება. განტოლება იქნება მოცემულ მონაცემთა მოდელი.

14. განვიხილოთ შემდეგი მაჩვენებლიანი განტოლება:

$$y = 3 \cdot 10^{2x}$$

თუ x გარკვეულ მნიშვნელობათათვის გამოვითვლით y მნიშვნელობებს და ავაგებთ x-y წრფივ-ლოგარითმულ გრაფიკს, წერტილები წრფეზე განლაგდებიან. ამაში რომ დავრწმუნდეთ ავიღოთ ორივე მხარის ლოგარითმი:

$$\log_{10} y = 2x + \log_{10} 3$$

ეს განტოლება წრფივია x და $\log_{10} y$ -თვის. MATLAB საშუალებით ააგეთ $y = 3 \cdot 10^{2x}$

გამოსახულების წრფივი და წრფივ-ლიგარითმული გრაფიკი. შეაფასეთ დახრა და y ღერძის გადაკვეთის წრეტილი. დახრა დახლოებით 2 უნდა იყოს, ხოლო y ღერძის გადაკვეთის წერტილი $\log_{10} 3$-ის ტოლი.

15. დავუშვათ გაქვთ ექსპერიმენტის შედეგად მიღებულ მონაცემთა მწკრივი, ააგეთ მისი წრფივ-ლოგარითმული გრაფიკი და დაახლოებით წრფე მიიღეთ. რა ტიპის განტოლება შეიძლება განვიხილოთ ასეთი მონაცემების მოდელად? როგორ გამოითვლით განტოლების მუდმივებს?

16. განვიხილოთ შემდეგი განტოლება:

$$y = 5x^3$$

თუ ავაგებთ ამ განტოლების ლოგარითმულ-ლოგარითმულ გრაფიკს, მივიღებთ წრფეს. იმიტომ, რომ თუ ავიღებთ განტოლების ორივე მხარის ლოგარითმს, მივიღებთ:

$$\log_{10} y = 3\log_{10} x + \log_{10} 5$$

ეს განტოლება კი წრფეს წარმოადგენს $\log_{10} x$ და $\log_{10} y$ -თვის. MATLAB საშუალებით ააგეთ $y = 5x^3$ განტოლების წრფივი და ლოგარითმულ-ლოგარითმული გრაფიკი. ამ უკანასკნეკის მიხედვით შეაფასეთ დახრა და y ღერძის გადაკვეთის წერტილი. დახრა დაახლოებით 3 –ის ტოლი უნდა მიიღოთ, y ღერძის გადაკვეთის წერტილი კი დაახლოებით $\log_{10} 5$ - ტოლი.

17. დავუშვათ გაქვთ ექსპერიმენტის შედეგად მიღებულ მონაცემთა მწკრივი, ააგეთ მისი ლოგარითმულ-ლოგარითმული გრაფიკი და დაახლოებით წრფე მიიღეთ. რა ტიპის განტოლება შეიძლება განვიხილოთ ასეთი მონაცემების მოდელად? როგორ გამოითვლით განტოლების მუდმივებს?

ფუნქცია sinc . ეს ფუნქცია საინტერესოა თავისი ფორმის გამო. მას აგრეთვე დიდი მნიშვნელობა აქვს ციფრული სიგნალის დამუშავებაში. ფუნქცია აღიწერება ფორმულით:

$$f(x, y) = \sin c(r) = \sin(r)/r$$

სადაც r (x,y) წერტილის მანძილია კოორდინატთა სათავიდან. სათავეში ფუნქციის მნიშვნელობა 1-ის ტოლია (ამ ფუნქციას, ფორმის გამო, ხშირად 'სომბრეროს' უწოდებენ).

18. შექმენით კვადრატული მატრიცა, რომლის ზომა იქნება (11 × 11), რომელიც შეიცავს sinc ფუნქციის მნიშვნელობებს ინტერვალში $-10 \le x \le 10$, $-10 \le y \le 10$. ააგეთ მიღებული ზედაპირის სამგანზომილებიანი გრაფიკი.
19. მე-18 ამოცანაში განხილული ზედაპირისთვის ააგეთ კონტურული გრაფიკი.
20. ააგეთ მე-18 ამოცანაში განხილული sinc ფუნქციის სამგანზომილებიანი გრაფიკი ხედვის კუთხით ზედაპირის ქვემოდან
21. ააგეთ მე-18 ამოცანაში განხილული sinc ფუნქციის სამგანზომილებიანი გრაფიკი ხედვის კუთხით დახრილობა = 0 გრადუსი.
22. ააგეთ sinc ფუნქციის კონტურული გრაფიკი კონტურებით დონეებზე: 0.1, 0.2, 0.3, 0.4, 05.