

استفاده از کامپیوتر در حل معادلات غیر خطی

می‌خواهیم مبحث آشنا و جالبی را مورد بحث و بررسی قرار دهیم و آن، استفاده از کامپیوتر در حل معادلات غیرخطی^۱ است. قبل از وارد شدن به بحث، نخست تعریفی از صفر^۲ یا ریشه^۳ یک معادله ارائه می‌دهیم.

تعریف: فرض کنید $y = f(x)$ یک تابع پیوسته^۴ باشد. هر عددی مانند x_0 که به ازای آن $f(x_0) = 0$ باشد یک ریشه یا یک صفر معادله^۵ $f(x) = 0$ نامیده می‌شود.

در بحثهایی که در زیر می‌آید فرض می‌کنیم تابع غیرخطی $y = f(x)$ در فاصله^۶ $[a, b]$ پیوسته بوده و $f(a) \cdot f(b) < 0$ است. به عبارت دیگر از فرض پیوسته بودن $y = f(x)$ و شرط $f(a) \cdot f(b) < 0$ ، نتیجه می‌گیریم که $f(x)$ در فاصله^۷ $[a, b]$ حداقل دارای یک ریشه است.

ریشه‌های هر معادله^۵ غیرخطی مانند $f(x) = 0$ را می‌توان به طور تقریبی از طریق ترسیم، روش تفکیک معادله^۸ $f(x) = 0$ به دو معادله و ترسیم آنها، یا با تشکیل جدولی از مقادیر متغیر^۹ و تابع به دست آورد که هیچ‌یک از آنها مورد نظر ما نیست. ما می‌خواهیم ریشه^{۱۰} تقریبی معادله^۵ غیرخطی $f(x) = 0$ را با استفاده از روشی موسوم به روش نیوتن یا به بیان دقیق‌تر با روش نیوتن - رفسون^{۱۱} به کمک یک برنامه^{۱۲} کامپیوتری^{۱۳} به دست آوریم. اینک شرح این روش:

در این روش، ابتدا کار را با یک تخمین^{۱۴} تقریبی^{۱۵} از ریشه^{۱۰} معادله که در فاصله^{۱۱} $[a, b]$ قرار دارد شروع می‌کنیم و تخمین اولیه^{۱۶} را x_0 می‌نامیم. پس از رسم نمودار $y = f(x)$ ، از نقطه^{۱۷} $(x_0, f(x_0))$ خطی مماس^{۱۸} بر منحنی^{۱۹} رسم می‌کنیم. نقطه تلافی^{۲۰} خط مماس بر منحنی و محور x ها، تخمین بعدی برای ریشه^{۱۰} معادله است. همین‌طور از نقطه^{۲۱} $(x_1, f(x_1))$ واقع بر منحنی، خط مماس دیگری بر منحنی رسم می‌کنیم. نقطه تلافی^{۲۲} خط مماس جدید و محور x ها، تخمین دیگری برای ریشه^{۱۰} معادله است والی آخر. اگر نقطه^{۲۳} $(x_n, f(x_n))$ آخرین نقطه^{۲۴} روی منحنی باشد که از آن نقطه^{۲۵} خط مماس بر منحنی رسم شده است و x_{n+1} آخرین تخمین برای ریشه^{۱۰} معادله باشد مقدار



● مهندس حسین ابراهیم زاده قلزم

$$\Rightarrow f'(x_n) = \frac{f(x_n)}{x_n - x_{n+1}}$$

$$\Rightarrow x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad \text{فرمول نیوتن-رفسون}$$

بنابراین با فرض اینکه $x_{n+1} = g(x)$ و $x_n = x$ باشد داریم:

$$g(x) = x - \frac{f(x)}{f'(x)}$$

در این روش همواره فرض می‌کنیم $f'(x) \neq 0$ است.

پس از این مقدمه، اکنون آماده‌ایم برنامه Y^0 ای کامیوتری برای به دست آوردن ریشه تقریبی هر معادله غیرخطی بنویسیم. اسیدوارم با ارائه چند مثال کامیوتری به همراه خروجی برنامه‌ها، شما را در درک بهتر روش نیوتن-رفسون یاری کرده باشم.

مثال: برنامه‌ای به زبان BASIC بنویسید تا با استفاده از روش

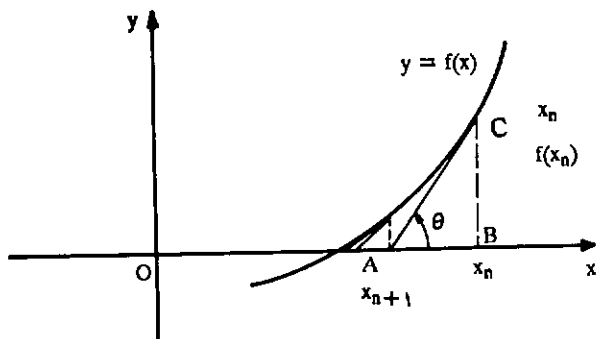
نیوتن-رفسون ریشه معادله $F(X) = X \sin X + \cos X = 0$

را که بین $[\pi, 0]$ قرار دارد با تقریب 10^{-6} به دست آورد. X بر حسب

رادیان است.

خطا^{۱۱} از رابطه $|x_{n+1} - x_n|$ به دست می‌آید. در این روش، در هر مرحله از تخمین ریشه، فاصله دو نقطه متوالی^{۱۷} روی محور X ها یا مقدار خطا را به دست می‌آوریم، اگر مقدار خطا از یک تقریب معینی مثلاً از 10^{-6} کمتر باشد آنگاه آخرین تخمین، x_{n+1} ، ریشه تقریبی معادله است.

طریقه به دست آوردن فرمول نیوتن-رفسون:



باتوجه به مثلث قائم‌الزاویه^{۱۸} ABC و تعریف مشتق^{۱۹} داریم:

$$\Delta ABC: \tan \theta = \frac{BC}{AB} = \frac{f(x_n)}{x_n - x_{n+1}} = f'(x_n)$$

برنامه:

```

10 REM NEWTON - RAPHSON METHOD FOR SOLVING NONLINEAR EQUATIONS
20 INPUT "ENTER A NUMBER BETWEEN 2 & 4 PLEASE : "; X
30 LET Y=X-(X*SIN(X)+COS(X))/(X*COS(X))
40 WHILE ABS(Y-X) > .000001
50     LET X = Y
60     LET FX = X*SIN(X)+COS(X)
70     LET FPRIME= X*COS(X)
80     LET Y = X-FX/FPRIME
90 WEND
100 PRINT "APPROXIMATE ROOT OF XSINX+COSX=0 EQUALS X= "; Y
110 END

```

خروجی^{۲۱} برنامه بالا با دو اجرای^{۲۲} مختلف به صورت زیر است:

```

ENTER A NUMBER BETWEEN 2 & 4 PLEASE : ? 2.3
APPROXIMATE ROOT OF XSINX+COSX=0 EQUALS X= 2.798386

```

اجرای اول:

اجرای دوم:

ENTER A NUMBER BETWEEN 2 & 4 PLEASE : ? 3.5
 APPROXIMATE ROOT OF XSINX+COSX=0 EQUALS X= 2.798386

همانگونه که ملاحظه می‌کنید ریشه تقریبی معادله داده شده، $X = 2.798386$ است.

مثال: برنامه‌ای به زبان BASIC بنویسید تا با استفاده از روش نیوتن-رفسون ریشه معادله $F(X) = X^3 + 3X + 1 = 0$ را که بین $[-1, 0]$ قرار دارد با تقریب 10^{-4} به دست آورد.

برنامه:

```
10 REM SOLVING X^3+3*X+1=0 BY NEWTON - RAPHSON METHOD
20 INPUT"ENTER A NUMBER : ";X
30 LET Y=X-(X^3+3*X+1)/(3*X^2+3)
40 WHILE ABS(Y-X) > .000001
50 LET X = Y
60 LET FX = X^3+3*X+1
70 LET FPRIME= 3*X^2+3
80 LET Y = X-FX/FPRIME
90 WEND
100 PRINT"X= ";Y;" IS A ROOT OF X^3+3*X+1=0 "
110 END
```

خروجی برنامه بالا با دو اجرای مختلف به صورت زیر است:

اجرای اول:

ENTER A NUMBER : ? -1
 X= -.3221854 IS A ROOT OF X^3+3*X+1=0

اجرای دوم:

ENTER A NUMBER : ? 20
 X= -.3221854 IS A ROOT OF X^3+3*X+1=0

همانگونه که ملاحظه می‌کنید ریشه تقریبی معادله داده شده، $X = -0.3221854$ است.

تمرین: اولاً نشان دهید^{۲۳} معادله $(2X + 1)^2 = 4\cos\pi X$ دارای یک ریشه در فاصله $[\frac{1}{4}, \frac{1}{3}]$ است، ثانیاً برنامه‌ای به زبان BASIC بنویسید تا با استفاده از روش نیوتن-رفسون ریشه تقریبی معادله $(2X + 1)^2 = 4\cos\pi X$ را تا چهار رقم اعشار^{۲۴} به دست آورد.

جواب: $X = -0.2872$

مثال: الف) با استفاده از فرمول نیوتن-رفسون، دستور محاسبه ریشه π ام^{۲۵} عدد ثابت مثبت و حقیقی C ^{۲۶} را به دست آورید. ب) برنامه‌ای به زبان BASIC بنویسید تا با خواندن^{۲۷} دو عدد N و C ، ریشه N ام عدد C را با تقریب 10^{-6} به دست آورد. برنامه را به صورت سؤال و جوابی^{۲۸} بنویسید تا هر بار سؤال کند که آیا داده دیگری دارید یا خیر؟

حل: الف) عدد حقیقی مثبت و ثابت C و عدد صحیح N داده شده است. ریشه Nام عدد C جواب معادله زیر است: $X^N = C$

$$F(X) = X^n - C$$

قرار می‌دهیم:

باتوجه به روش نیوتن - رفسون و بافرض اینکه $X_{n+1} = Y$ و $X_n = X$ باشد داریم:

$$Y = X - \frac{X^N - C}{NX^{N-1}}$$

$$= \left[(N-1) \cdot X + \frac{C}{X^{N-1}} \right] / N$$

مثلاً به ازای $N = 3$ ، برای ریشه سوم عدد C، فرمول زیر را داریم:

$$Y = \left[2X + \frac{C}{X^2} \right] / 3$$

ب - برنامه:

```

10  REM THE Nth ROOT OF A CONSTANT C
20  INPUT"DO YOU WANT THE PROGRAM EXECUTE : ";ANSS
30  WHILE ANSS="Y" OR ANSS="y"
40      INPUT"ENTER INTEGER & REAL NUMBERS ";C,N
50      INPUT"ENTER YOUR FIRST GUESS ";X
60      Y= ((N-1)*X+C / X^(N-1)) / N
70      WHILE ABS(Y-X) > .000001
80          LET X = Y
90          LET Y = ((N-1)*X+C/X^(N-1)) / N
100     WEND
110     PRINT"THE ";N;"th ROOT OF ";C;" =";Y
120     PRINT
130     INPUT "DO YOU HAVE ANY MORE DATA ";ANSS
140     WEND
150     PRINT "THE PROGRAM HAS ENDED"

```

خروجی برنامه بالا پس از اجرا به صورت زیر است:

```

DO YOU WANT THE PROGRAM EXECUTE : ? Y
ENTER INTEGER & REAL NUMBERS ? 2,2
ENTER YOUR FIRST GUESS ? 1
THE 2 th ROOT OF 2 = 1.414214

```

```

DO YOU HAVE ANY MORE DATA ? Y
ENTER INTEGER & REAL NUMBERS ? 3,2
ENTER YOUR FIRST GUESS ? 1.1
THE 2 th ROOT OF 3 = 1.732051

```

DO YOU HAVE ANY MORE DATA ? Y
 ENTER INTEGER & REAL NUMBERS ? 5,3
 ENTER YOUR FIRST GUESS ? 2.2
 THE 3 th ROOT OF 5 = 1.709976

DO YOU HAVE ANY MORE DATA ? Y
 ENTER INTEGER & REAL NUMBERS ? 7,5
 ENTER YOUR FIRST GUESS ? 3.5
 THE 5 th ROOT OF 7 = 1.475773

DO YOU HAVE ANY MORE DATA ? N
 THE PROGRAM HAS ENDED

تمرین: برنامه‌ای به زبان BASIC بنویسید تا با استفاده از روش نیوتن -رفسون، ریشه مثبت^{۲۹} معادله $2 \sin X = X$ را که بین $[0, 2]$ قرار دارد تا چهار رقم اعشار به دست آورد.

جواب: $X = 1.8955$

تمرین: برنامه‌ای به زبان BASIC بنویسید تا با استفاده از روش نیوتن -رفسون ریشه تقریبی معادله $F(X) = X^3 + X - 1 = 0$ را که در فاصله $[0, 1]$ قرار دارد^{۳۰} با تقریب 10^{-6} به دست آورد.

جواب: $X = 0.682328$

استفاده از کامپیوتر در تعیین نقاط ماگزیموم و مینیموم یک تابع

فرض کنید در فاصله $[a, b]$ در جستجوی^{۳۱} دو مقدار برای متغیر x هستیم که به ازای آنها مقدار تابع $y = f(x)$ ماگزیموم و مینیموم می‌شود. به ازای x هایی که در فاصله $[a, b]$ قرار دارند مقادیر مختلفی برای $y = f(x)$ به دست می‌آید و ما می‌خواهیم کوچکترین^{۳۲} و بزرگترین^{۳۳} مقدار y را در بین این مقادیر به دست آوریم.

قبل از نوشتن برنامه تعیین نقاط ماگزیموم و مینیموم یک تابع، الگوریتم^{۳۴} برنامه‌ای را می‌نویسیم که با دریافت N ^{۳۵} عدد، کوچکترین و بزرگترین عدد داخل لیست N عددی را پیدا کرده در خروجی چاپ می‌کند.

الگوریتم تعیین کوچکترین و بزرگترین عدد یک لیست N عددی به شرح زیر است:

۰ - شروع

۱ - عدد N را می‌خوانیم.

۲ - اولین عدد را خوانده آن را NUM می‌نامیم.

۳ - مقدار $COUNT$ را به عنوان شماره^{۳۶} تعداد اعداد برابر 1 قرار می‌دهیم^{۳۷}.

۴ - مقدار MAX و MIN را برابر NUM قرار می‌دهیم.

۵ - عدد بعدی را خوانده آن را برابر NUM قرار می‌دهیم.

۶ - به مقدار $COUNT$ یک واحد اضافه کرده آن را $COUNT$ می‌نامیم.

- ۷- اگر مقدار MAX کوچکتر از مقدار NUM باشد مقدار NUM را برابر MAX قرار می‌دهیم.
- ۸- اگر مقدار MIN بزرگتر از مقدار NUM باشد مقدار NUM را برابر MIN قرار می‌دهیم.
- ۹- اگر مقدار COUNT برابر مقدار N نبود به مرحله ۵ می‌رویم.
- ۱۰- مقدار MAX و MIN را به عنوان کوچکترین و بزرگترین عدد داخل لیست با پیغام^{۳۸} مناسب چاپ می‌کنیم.

۱۱- پایان

در برنامه زیر روش کار به همان صورتی است که در الگوریتم بالا گفته شده است با این تفاوت که اعداد هر بار توسط تابع تولید^{۳۹} می‌شوند و پایان داده^{۴۰}ها هم مقدار معلوم $f(b)$ است و از این رو در برنامه احتیاج به شمارنده نداریم. با یک حلقه^{۴۱} FOR برنامه براحتی قابل پیاده‌سازی^{۴۲} است.

مثال: برنامه‌ای به زبان BASIC بنویسید تا ماگزیموم و مینیموم تابع $Y = \sin X + \cos X$ را در فاصله $[0, 2\pi]$ به دست آورده، مقادیر ماگزیموم و مینیموم تابع را به همراه Xهای مربوطه^{۴۳} با پیغام مناسب در خروجی چاپ کند.

```

10 REM FINDING MAX & MIN OF Y=SINX+COSX
20 LET MAX=0: LET MIN=0
30 FOR D = 0 TO 360
40 LET R = 4*ATN(1)*D/180
50 LET Y = SIN(R)+COS(R)
60 IF MAX < Y THEN MAX = Y : LET K=D
70 IF MIN > Y THEN MIN = Y : LET S=D
80 NEXT D
90 PRINT "MAXIMUM = ";MAX;"AT X= ";K;" DEGREES"
100 PRINT
110 PRINT "MINIMUM = ";MIN;"AT X= ";S;"DEGREES"
120 END

```

برنامه:

خروجی برنامه:

MAXIMUM = 1.414214 AT X= 45 DEGREES

MINIMUM = -1.414214 AT X= 225 DEGREES

مثال: برنامه‌ای به زبان BASIC بنویسید تا ماگزیموم و مینیموم تابع $Y = 2 \sin X + 3 \cos X$ را در فاصله $[0, 2\pi]$ به دست آورده، مقادیر ماگزیموم و مینیموم تابع را به همراه Xهای مربوطه با پیغام مناسب در خروجی چاپ کند.

برنامه: این برنامه دقیقاً مانند برنامه $Y = \sin X + \cos X$ است فقط با این تفاوت که لازم است خط 10 برنامه مثال قبل را به صورت زیر تغییر

```

10 REM FINDING MAX & MIN OF Y = 3*SINX+3*COSX

```

دهیم:

```

50 LET Y = 2*SIN(R) + 3*COS(R)

```

و در خط 50، ضابطه تابع جدید را به صورت زیر بنویسیم:

با اعمال تغییر در خط 10 و 50 برنامه مثال قبل، پس از اجرا، خروجی برنامه را به صورت زیر مشاهده خواهید کرد:

MAXIMUM = 3.605499 AT X= 34 DEGREES

MINIMUM = -3.605498 AT X= 214 DEGREES

مثال: برنامه‌ای به زبان BASIC بنویسید تا ماگزیوموم و مینیوموم مطلق^{۴۴} تابع $y = x^3 - x^2 - 120x - 20$ را در فاصله [-10, 10] به دست آورده، نتیجه را با پیغام مناسب در خروجی چاپ کند.

برنامه:

```
10 REM FINDING MAX & MIN OF Y=X^3-X^2-120*X-20
20 LET MAX=0: LET MIN=0
30 FOR X = -10 TO 10
40 LET Y = X^3-X^2-120*X-20
50 IF MAX < Y THEN MAX = Y : LET K=X
60 IF MIN > Y THEN MIN = Y : LET S=X
70 NEXT X
80 PRINT "MAXIMUM OF Y= ";MAX;"AT X= ";K
90 PRINT
100 PRINT "MINIMUM OF Y= ";MIN;"AT X= ";S
110 END
```

خروجی برنامه:

MAXIMUM OF Y= 448 AT X= -6

MINIMUM OF Y= -566 AT X= 7

مثال: برنامه‌ی مثال قبل را به گونه‌ای تغییر دهید تا ماگزیوموم و مینیوموم مطلق تابع $y = x^3 - x^2 - 120x - 20$ را در فاصله [-13, 13] به دست آورد.

برنامه: در این برنامه کافی است خط شماره 30 مثال قبل را به صورت زیر تغییر دهیم:

```
30 FOR X = -13 TO 13
```

با اعمال تغییر بالا در خط 30 از مثال قبل، پس از اجرا، خروجی برنامه را به صورت زیر مشاهده خواهید کرد:

MAXIMUM OF Y= 448 AT X= -6

MINIMUM OF Y= -826 AT X= -13

مثال: برنامه‌ای به زبان BASIC بنویسید تا ماگزیوموم و مینیوموم نسبی^{۴۵} تابع $y = 2x^3 - 3x^2 - 12x + 7$ را در فاصله [-10, 10] به دست آورده، نتیجه را با پیغام مناسب در خروجی چاپ کند.

برنامه:

```
10 REM FINDING MAX & MIN OF Y=2*X^3-3*X^2-12*X+7
20 FOR X = -10 TO 10
30 LET Y = 2*X^3-3*X^2-12*X+7
40 LET X1 = X-.01
50 LET X2 = X+.01
60 LET Y1 = X1^2-X1-12
70 LET Y2 = X2^2-X2-12
80 IF Y1*Y2 < 0 AND Y1 < 0 THEN PRINT"AT X= ";X;" Y= ";Y;" IS MINIMUM"
90 IF Y1*Y2 < 0 AND Y1 > 0 THEN PRINT"AT X= ";X;" Y= ";Y;" IS MAXIMUM"
100 NEXT
110 END
```

خروجی برنامه:

AT X= -3 Y= -38 IS MAXIMUM
 AT X= 4 Y= 39 IS MINIMUM

مثال: برنامه‌ای به زبان BASIC بنویسید تا ضرایب یک تابع درجه سوم، کران پایین و بالای یک فاصله را از ورودی بخواند. مقادیر ماکزیموم و مینیموم نسبی و مطلق تابع را در آن فاصله به دست آورده، نتیجه را با پیغام مناسب در خروجی چاپ کند.

برنامه:

```

10 PRINT"Y=AX^3+BX^2+CX+D"
20 INPUT"Enter A,B,C,D";A,B,C,D
30 LOCATE 3,1: INPUT"PLEASE ENTER L & U [L,U]" ; L , U
40 IF U < L THEN PRINT"U SHOULD BE GREATER THEN L ":GOTO 40
50 OLDY = A*(L-.01)^3+B*(L-.01)^2+C*(L-.01)+D
60 YMAX= OLDY:YMIN=OLDY:FALSE1=0:
70 FOR X = L TO U STEP .05
80     NEWY = A*X^3+B*X^2+C*X+D
90     IF YMAX < NEWY THEN YMAX=NEWY
100    IF YMIN > NEWY THEN YMIN=NEWY
110    IF NEWY > OLDY THEN FALSE = 1 ELSE FALSE = -1
120    OLDY=NEWY
130    IF FALSE1-FALSE = 2 THEN PRINT"AT X = "X-.049:PRINT"REL MAX OF Y="OLDY
140    IF FALSE1-FALSE = -2 THEN PRINT"AT X = "X-.049:PRINT"REL MIN OF Y="OLDY
150    FALSE1=FALSE
160 NEXT
170 PRINT"ABSOLUTE MAXIMUM OF Y=";YMAX
180 PRINT"ABSOLUTE MINIMUM OF Y=";YMIN
    
```

اجرای اول:

اجرای دوم:

Y=AX^3+BX^2+CX+D
 Enter A,B,C,D? 2,-6,0,5
 PLEASE ENTER L & U [L,U]? -4,4
 AT X = 9.975508E-04
 REL MAX OF Y= 4.985252
 AT X = 2.000997
 REL MIN OF Y=-2.984753
 ABSOLUTE MAXIMUM OF Y= 36.99976
 ABSOLUTE MINIMUM OF Y=-220.443

Y=AX^3+BX^2+CX+D
 Enter A,B,C,D? 1,0,-3,3
 PLEASE ENTER L & U [L,U]? -4,4
 AT X = -.9990026
 REL MAX OF Y= 4.992626
 AT X = 1.000998
 REL MIN OF Y= 1.007624
 ABSOLUTE MAXIMUM OF Y= 54.99978
 ABSOLUTE MINIMUM OF Y=-49.45121

واژه‌نامه ریاضی و کامپیوتر

- | | | |
|-----------------------------|-----------------------------|--------------------|
| 1 - Nonlinear | 13 - A tangent Line | 25 - The nth Root |
| 2 - Zero | 14 - Curve | 26 - Real |
| 3 - Root | 15 - Intersection | 27 - Read |
| 4 - Continuous Function | 16 - Error value | 28 - Interactive |
| 5 - Equation | 17 - Two successive points | 29 - Positive Root |
| 6 - Variable | 18 - Right Triangle | 30 - Lies |
| 7 - Newton - Raphson method | 19 - Derivative | 31 - Search |
| 8 - A computer program | 20 - Program | 32 - Smallest |
| 9 - Estimate | 21 - output | 33 - Largest |
| 10 - Approximate | 22 - Running | 34 - Algorithm |
| 11 - interval | 23 - Show That | 35 - Get |
| 12 - Initial Estimate | 24 - To Four Decimal Places | 36 - Counter |
| | | 37 - Set |
| | | 38 - Message |
| | | 39 - Generate |
| | | 40 - End of Data |
| | | 41 - Loop |
| | | 42 - Implementable |
| | | 43 - Corresponding |
| | | 44 - Absolute |
| | | 45 - Relative |