درس ۳۱

حل یک مدل برنامه ریزی خطی ساده

تهیه شده توسط گروه آموزشی بهینه‌یاب

BEHINEHYAB.COM

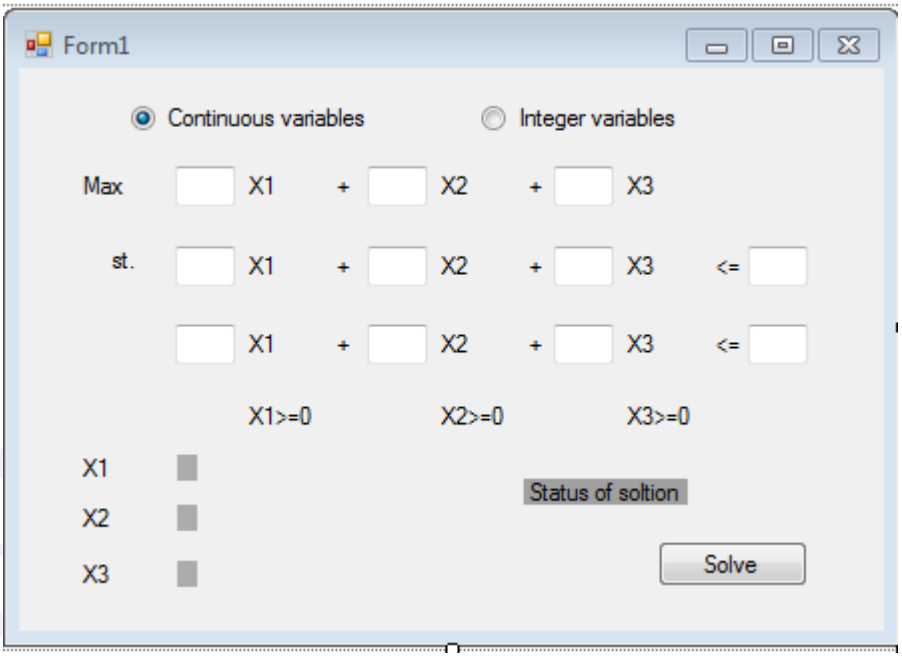
مقدمه

در این بخش به حل یک مسئله برنامه ریزی خطی با متغیرهای پیوسته و گسسته با استفاده از کتابخانه *cplex* در محیط ویژوال بیسیک می پردازیم. در این مثال، مقادیر ضرایب تابع هدف و محدودیت ها و مقادیر سمت راست از کاربر گرفته می شود. نوع متغیرها هم در اختیار کاربر است که پیوسته باشد یا گسسته.

مدل مدنظر به صورت زیر است:

$$
\begin{aligned}
&Max \quad c_1x_1+c_2x_2+c_3x_3\\
&s.t.\\
&a_{11}x_1+a_{12}x_2+a_{13}x_3 \le b_1\\
&a_{21}x_1+a_{22}x_2+a_{23}x_3 \le b_2\\
&x_1,x_2,x_3 \ge 0
\end{aligned}
$$

برای اغاز کار، یک پروژه جدید با نام *Exercise3* ایجاد می کنیم و فرم زیر را ایجاد می کنیم.



در فرم فوق، ضرایب متغیرها و مقادیر به صورت *textbox* از کاربر گرفته می شود. همچنین نوع متغیرها(پیوسته یا گسسته) به صورت *radiobutton* از کاربر گرفته می شود. پس از ورود اطلاعات، بر روی دکمه *solve* کلیک می کنیم. در صورتی که مدل جواب داشته باشد، مقادیر بهینه متغیرها روبروی متغیرها نمایش داده می شود. در غیر این صورت برنامه خطای امکان ناپذیر می دهد.

برای وارد کردم برنامه، بر روی دکمه *solve* دوبار کلیک می کنیم و کد زیر را وارد می کنیم. با توجه به اهمیت این مثال در نحوه استفاده از قابلیت های سیپلکس در ویژوال بیسیک، به صورت دقیق هر خط توضیح داده می شود.

در این بخش تعریف مدل، متغیرهای و محدودیت ها انجام می شود که از کتابخانه *cplex* به صورت مستقیم انجام می شود.	``` Dim cplex As New Cplex() Dim var(1)() As INumVar Dim rng(1)() As IRange Dim x(2) As INumVar ```

در صورتیکه کاربرد متغیر را پیوسته در نظر بگیرید، تعریف متغیر عبارت *float* نوشته می شود. همچنین می توان حد پایین و بالا متغیر را در این بخش تعریف کرد که در این جا به ترتیب ۰ و ۳۳ است. در صورتیکه کاربرد متغیر را گسسته انتخاب نماید، نوع متغیر به صورت *int* یا عدد صحیح انتخاب می شود.	```
	If rbtncontin.Checked = True Then
	For i = 0 To UBound(x, 1)
	x(i) = cplex.NumVar(0, 33,
	ILOG.Concert.NumVarType.Float)
	
	Next
	
	End If
	If rbtninteger.Checked = True Then
	For i = 0 To UBound(x, 1)
	x(i) = cplex.NumVar(0, 33,
	ILOG.Concert.NumVarType.Int)
	
	Next
	
	End If
	``

در صورتیکه چند نوع متغیر وجود داشته باشد، استفاده از متغیر *var* کمک می کند . به این صورت که هر گونه در یک در یکی از درایه ها ذخیره می شود. برای مثال اگر متغیر *y* هم وجود داشته باشد، می توان این متغیر را در *var(1)* ذخیره کرد. با این کار در گزارش نتایج کمک می کند. ضرایب متغیرها در تابع هدف در بردار *objvalus* قرار می گیرد که از کاربر گرفته می شود. برای تعریف محدودیت ها و تابع هدف، نیاز به تعریف *expression* است که از کتابخانه *cplex* کمک گرفته می شود.	``` var(0) = x Dim objvals As Double() = {CDbl(txtobjX1.Text), CDbl(txtobjX2.Text), CDbl(txtobjX3.Text)} Dim experssion(2) As ILOG.Concert.ILinearNumExpr Dim objfunction As ILOG.Concert.IObjective = cplex.AddMaximize() ```

BEHINEHYAB.COM

بیشینه سازی تابع هدف هم در این بخش تعریف می شود.	
تابع هدف در مولفه اول *expressioin* ذخیره می شود و سپس به مدل با دستور *addtoexpre* اضافه می شود.	``` experssion(1) = cplex.ScalProd(x, objvals) cplex.AddToExpr(objfunction, experssion(1)) ```
در این بخش محدودیت ها تعریف می شود. در صورتیکه محدودیت به صورت بزرگتر مساوی باشد، از عبارت *addge* استفاده می شود. به این صورت مدل به صورت کامل وارد می شود.	``` rng(0) = New IRange(1) {} rng(0)(0) = cplex.AddLe(cplex.Sum(cplex.Prod(CInt(txtCon1X1.Text), x(0)), cplex.Prod(CInt(txtCon1X2.Text), x(1)), cplex.Prod(CInt(txtCon1X3.Text), x(2))), CInt(txtCon1B.Text), "c1") rng(0)(1) = cplex.AddLe(cplex.Sum(cplex.Prod(CInt(txtCon2X1.Text), x(0)), cplex.Prod(CInt(txtCon2X2.Text), x(1)), cplex.Prod(CInt(txtCon2X3.Text), x(2))), CInt(txtCon2B.Text), "c2") ```

اکنون مدل امده حل است. برای این منظور از دستور *cplex.solve* استفاده می شود. مقدار این تابع به صورت *true* و *false* است. در صورتیکه مدل جواب داشته باشد، مقدار*true* می گیرد. در غیر این صورت، مسئله جواب ندارد و برنامه عبارت
No optimal solution found
چاپ می شود. در صورتیکه مدل دارای جواب باشد، با دستورهای *getvalues* می توان مقدار بهینه متغیرها را بدست اورد. وضعیت جواب توسط دستور *getstatus* گزارش می شود که در صورت بهینگی، مقدار آن *optimal* خواهد بود. مقدار تابع هدف بهینه از دستور *cplex.objvalue.tostring* به دست می آید. مقدار بهینه متغیرها در بردار*xx* ذخیره شده است که گزارش می شود.
	``
	   If cplex.Solve() Then
	Dim xx As Double() = cplex.GetValues(var(0))
	cplex.Output().WriteLine(("Solution status = " +
	cplex.GetStatus().ToString))
	            lblStatus.Text =
	cplex.GetStatus().ToString
	            cplex.Output().WriteLine(("Solution value
	= " + cplex.ObjValue.ToString))
	
	lblX1.Text = xx(0)
	lblX2.Text = xx(1)
	lblX3.Text = xx(2)
	MessageBox.Show(("Solution value = " +
	cplex.ObjValue.ToString))
	
	Else
	MessageBox.Show(("No optimal solution found"))
	End If
	
	If cplex.GetStatus().ToString = "Optimal" Then
	MessageBox.Show("Optimal")
	End If
	cplex.End()
	``

BEHINEHYAB.COM

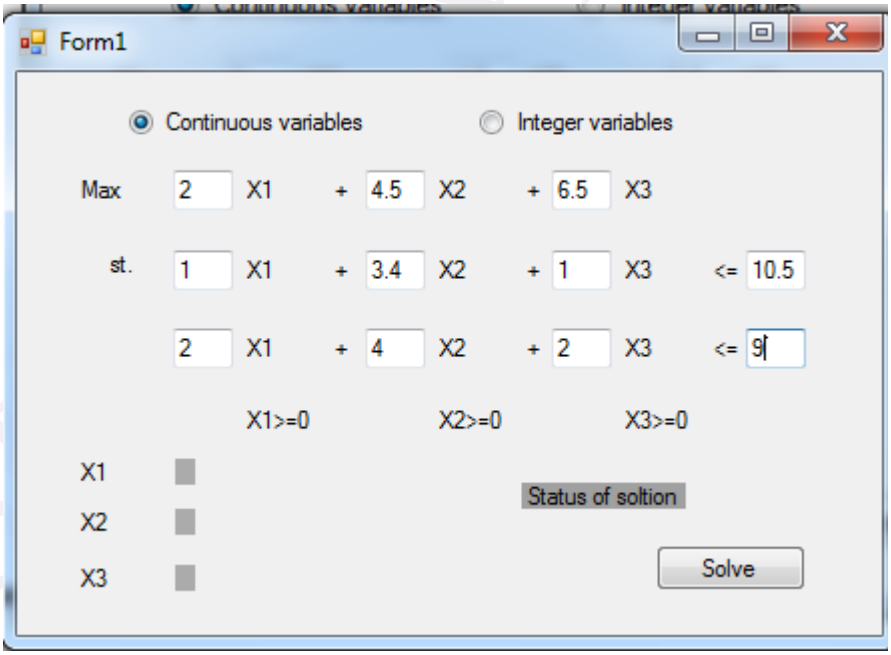
پـــس از پایـــان هـــر مـــدل، عبـــارت *cplex.End()* استفاده شود که فضای مورد استفاده از کـامپیوتر ازاد شود.	

برای نمونه چند مثال حل می کنیم.

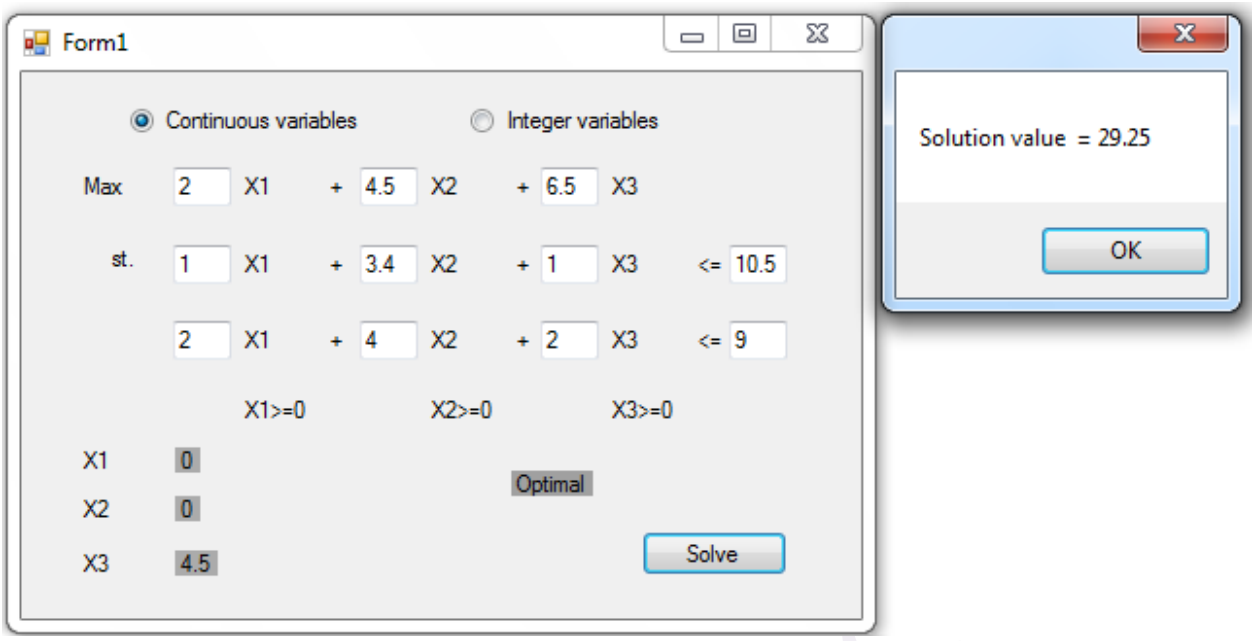
مدل زیر را در نظر بگیرید:

$$
\begin{aligned}
&Max \quad 2x_1 + 4.5x_2 + 6.5x_3 \\
&s.t. \\
&1x_1 + 3.4x_2 + 1x_3 \le 10.5 \\
&2x_1 + 4x_2 + 2x_3 \le 9 \\
&x_1, x_2, x_3 \ge 0
\end{aligned}
$$

ورود مدل فوق در فرم برنامه به صورت زیر می شود.



جواب بهینه مدل فوق به صورت زیر می شود.



در صورتیکه متغیرهای این مدل، گسسته باشد، جواب به صورت زیر می شود.