تقرير مادة بحوث العمليات

تقديم الطلاب

عبد الهادي بيطار

محمد عمار الشرفاوي الجزائرلي

محمد وسام الرجولة

# سمبلكس الثنوية:

صمم الكود للتعامل مع مسائل القيمة العظمى و قيود من النمط أصغر أو يساوي و متغيرات أكبر من الصفر لكن بالطبع يمكن تحويل أي مسألة للنمط المذكور

نلاحظ في الجداول اليدوية و قواعد إيجاد الحل الأمثل بالسمبلكس الثنوية أن أمثال المتغيرات الأساسية دائماً مساوية للصفر في دالة الهدف و تشكل المصفوفة الواحدية في أسطر القيود لذلك نستغني عن حفظها و القيام بعمليات مباشرة عليها في البرنامج المطلوب

بالتالي نحفظ أمثال المتغيرات الغير أساسية في دالة الهدف و القيود بالإضافة إلى قيم الطرف الأيمن في مصفوفة واحدة (table)وللتأكد من عدم ضياع مواقع المتغيرات نحفظ الأساسية في مصفوفة (variables)و الغير أساسية في مصفوفة ثانية (slacks)

|  |  |  |  |
| --- | --- | --- | --- |
| RH | Variable[1] | Variable[0] |  |
| Table[0,2] | Table[0,1] | Table[0,0] | Z |
| Table[1,2] | Table[1,1] | Table[1,0] | Slack[0] |
| Table[2,2] | Table[2,1] | Table[2,0] | Slack[1] |

**findOptimal**

تقوم هذه التعليمة بالانتقال إلى الجدول النهائي بالخوارزمية التالية

يوجد حل

لا يوجد حل

الحل مقبول للأولية

الحل أمثلي

الحل محدود

أوجد المتغير الخارج

أوجد المتغير الداخل الموافق للخارج

أخرج الأول و أدخل الثاني

الحل مقبول للثانوية

أوجد المتغير الداخل

أوجد المتغير الخارج الموافق للداخل

أدخل الأول و أخرج الثاني

نعم

نعم

لا

لا

نعم

نعم

لا

لا

يكون الحل مقبول للأولية إذا كانت قيمة الطرف الأيمن لكل القيود موجبة

يكون الحل أمثلياً عندما أمثال المتغيرات الأساسية موجبة

يكون الحل محدود إذا كان لكل متغير قابل للإدخال (قيمته سالبة) متغير واحد خارج على الأقل

يكون الحل مقبول للثانوية إذا كان لكل متغير قابل للإخراج (قيمة الطرف الأيمن للقيد المقابل له أصغر من الصفر) متغير واحد داخل على الأقل

## إيجاد المتحول الداخل للأولية:

هو صاحب أصغر أمثال سالبة

## إيجاد المتحول الخارج للأولية:

هو المتغير المتعلق بالقيد الذي ناتج قسمة القسم الأيمن على أمثال المتغير الداخل الموجبة الأصغر

## إيجاد المتحول الخارج للثانوية:

هو المتغير المتعلق بالقيد الذي قسمه الأيمن أصغر من الباقي

## إيجاد المتحول الداخل للثانوية:

هو المتغير الذي ناتج قسمة أمثاله في دالة الهدف على أمثاله السالبة فقط في القيد المتعلق بالمتغير الداخل الأصغر

## الإدخال و الإخراج:

أولا نحتفظ بقيمة أمثال المتغير الداخل في قيد المتغير الخارج (factor)

سيحل محل عمود المتغير الداخل عمود المتغير الخارج نوجده و نحتفظ به

نقسم سطر المتغير الخارج على factor

ثم نجري التحويلات السطرية و نضع عمود المتغير الخارج في مكانه

ونتأكد من عدم ضياع المتحول الداخل و الخارج بتبديل موضعه في slacks , variables

# خوارزمية النقل:

صمم الكود للتعامل مع مسائل حساب الكلفة الأدني و الربح الأعظم عندما الطلب يساوي الإنتاج

لسهولة الشرح سيتم شرح الكلفة الأدنى

تم حفظ الكلف في مصفوفة و الطلب في مصفوفة آخرى و كذلك الإنتاج و متغيرات النقل

**findOptimal**

تقوم هذه التعليمة بالانتقال إلى الجدول النهائي بالخوارزمية التالية

يوجد حل

لا يوجد حل

الحل الأولي مقبول

الحل أمثلي

أوجد الحل التالي

نعم

لا

لا

نعم

يكون الحل الأولي مقبول إذا كان الطلب يساوي تماماً الإنتاج

يكون الحل أمثلياً عندما قيمة أصغر (~c[i,j] = c[i,j] – Ui – Vj)أكبر أو تساوي الصفر

## إيجاد أصغر أو أكبر ~c:

بالتابع findMinCBar

### طريقة U V

CBarFinder

لدينا عدد المنتجين un عدد المستوردين vn مصفوفة الكلف c و مصفوفة المتغيرات نوجد الحل باستخدام SOLVE

بما أنه عدد المتغيرات الفعلي أصغر من مجموع المنتجين مع المستوردين ب 1

نفرض V0=0 دائما

التابعان العوديان findU , findV يستدعي كل منهما الآخر على الشكل التالي:

عند استدعاء findU يمر على العمود index إذا كان Uغير موجود و المتغير له قيمة يحسب U الموافق لها و يستدعي findV عند سطر U

عند استدعاء findV يمر على السطر index إذا كان Vغير موجود و المتغير له قيمة يحسب V الموافق لها و يستدعي findU عند العمود V

بهذه الطريقة و لأنه لابد من وجود قيمة لمتغير واحد على الأقل في كل سطر أو عمود نحصل على U V بعدد قليل من المسح للمصفوفة ثنائية البعد الذي يعكس سرعة في إيجاد الحل الأمثل

بعد إيجاد U V أصبح من السهل إيجاد أصغر ~c وأكبر ~c باستخدام (~c[i,j] = c[i,j] – Ui – Vj) بالمرور على المصفوفة

## إيجاد الحل التالي pivote

بما أنه لدينا موقع أصغر ~c يجب إيجاد الحلقة الموافقة لها

يتم إيجاد الحلقة باستخدام search

لنكن في الموقع I,j نبحث عن المواقع المحتملة التالية في السطر أو العمود الذي نحن فيه ثم من أجل كل موقع جديد نحفظه في المسار و نبحث عن المواقع المحتملة التالية في السطر إذا كنا قد بحثنا في العمود أو نبحث في العمود إذا كنا قد بحثنا في السطر إلى أن نعود إلى أول نقطة و في المسار أكثر من 3 مواقع

بعدما حصلنا على المسار يجب حساب القيمة الجديدة للمتغير الداخل إلى الحل و نجدها من المسار باستخدام findMu الذي يجلب أكبر قيمة ممكنة ثم نعدل على المسار و نخرج أحد المتغيرات و في حال وجد أكثر من متغير يمكنه الخروج نخرج صاحب الكلفة الأكبر

## إيجاد الحل الابتدائي

### طريقة الزاوية الشمالية الغربية:

التابع public static double[][]NorthWestCorner(double[][] c,double []a,double []b)

C مصفوفة الكلف a مصفوفة الإنتاج b مصفوفة الطلب و يجب أن يكون الطلب مساوياً للإنتاج

يحدد الموقع باستخدام i j نبدأ من الزاوية اليسرى الشمالية 0 0

إذا كان الإنتاج أصغر من الطلب المتغير يساوي الإنتاج و ينقص الطلب بمقدار الإنتاج و ننتقل للسطر التالي i++

و إلا فإن الطلب هو الأصغر فينقص الإنتاج و المتغير يساوي الطلب و ننتقل للعمود التالي j++ وفي حال تساوى الطلب مع الإنتاج فإننا ننتقل أيضاً للسطر التالي

### طريقة الكلفة الأدنى:

public static double[][]MinimomCoast(double[][] c,double []a,double []b)

C مصفوفة الكلف a مصفوفة الإنتاج b مصفوفة الطلب و يجب أن يكون الطلب مساوياً للإنتاج

يحدد الموقع باستخدام i j نبدأ من الزاوية اليسرى الشمالية 0 0 ونقوم بمسح الجدول للحصول على القيمة الأدنى

وذلك بشرط ان تكون القيمة الصغرى لا يقابلها مستودع فارغ او سوق اكتفى

ونقوم بهذا المسح حتى لا تبقى قيم ليتم اضافتها الى مصفوفة الحل

وفي كل عملية حصول على ادنى قيمة بالجدول

نقوم بإضافة القيمة الموافقة ل الأصغر بين المستودع والسوق الى مصفوفة الحل  
وطرح القيمة الأصغر من الأكبر بينهما