تصنيف أنصاف الزمر من المرتبة الثالثة ومن المرتبة الرابعة

حازم طلب[1] وعبد الواحد أبو حمدة[2]

تاريخ الإيداع 2013/10/31
قبل للنشر في 2014/03/03

الملخص

قدّمتُ في هذه الورقة البحثية طريقة جديدة لدراسة أنصاف الزمر من المرتبة الثالثة ومن المرتبة الرابعة، بُغيةَ استعراضها وتقديم وصف أفضل لها، وذلك لملاحظتنا الازدياد الكبير بتعداد أنصاف الزمر من مراتب أعلى.

قدّمتُ كذلك الطرائق التي كتبناها واتبعناها في دراسة الخاصة التجميعية وخلال عملية التصنيف.

الكلمات المفتاحية: أنصاف الزمر، أنصاف الزمر المنتهية، تصنيف أنصاف الزمر، خوارزميات أنصاف الزمر، تعداد أنصاف الزمر.

رقم التصنيف الرياضياتي: MSC 20M99

[1] طالب ماجستير، [2] أستاذ، قسم الرياضيات، كلية العلوم، جامعة دمشق، سورية.

Classification of the semigroups of order three and four

H. Talab[1] and A. Abohamdah[2]

Received 31/10/2013
Accepted 03/03/2014

ABSTRACT

In this paper we are presenting a new method for studying the semigroups of order three and four. We are going to explore these semigroups and illustrate a better description of them while observing their enormous increasing in number for larger orders.

We are presenting the methods we wrote and followed during our study of associativity and during the classification process.

Keywords: Semigroups, Finite semigroups, Semigroups classification, Semigroups algorithms, Semigroups enumeration.

Mathematics Subject Classification: 20M99

[1] MCS., Student, [2] Professor, Department of Mathematics, Faculty of Sciences, Damascus University, Syria

1. تمهيد

تعرَّف **البنية الجبرية (groupoid)** بأنها مجموعة غير خالية مزودة بقانون تشكيل داخلي[1]. على سبيل المثال إن كلاً من الزمرة والحقل والفضاء الشعاعي هي من البنى الجبرية الشائعة التي لها تطبيقات في الهندسة والعلوم والصناعة والاقتصاد. وإن فهماً أعمق لهذه البنى بحد ذاتها سيقودنا إلى فهم أعمق لاستخدامها ولتطبيقاتها.

تعدُّ نصف الزمرة واحدةً من أبسط هذه البنى الجبرية المثيرة للاهتمام، وإن بساطة تعريف نصف الزمرة وتَضمُّنها في عدد من البنى الأخرى يثيران في الذهن العديد من التساؤلات للبحث والتفكير فيها.

2. مقدمة

تعدُّ نصف الزمرة تعميماً للزمرة التي هي أحد مفاهيم الجبر المجرد، ولنظرية أنصاف الزمر المنتهية أهمية خاصة في علوم الحاسوب؛ وذلك لارتباطها الوثيق بنظرية الأتومات المنتهية Finite Automata، وقد تساءل الرياضيون منذ القرن التاسع عشر عن إمكانية تصنيف الزمر البسيطة المنتهية، ولم يصل هؤلاء إلى الهدف المنشود إلا في مطلع الثمانينيات من القرن الماضي، إذْ تطلب ذلك كتابة العديد من الصفحات، حتى أنهم درسوا الزمر المنتهية من الرتب العالية ومنها زمرة "الوحش" (Monster group) التي مرتبتها

$$2^{46} \times 3^{20} \times 5^{9} \times 7^{6} \times 11^{2} \times 13^{3} \times 17 \times 19 \times 23 \times 29 \times 31 \times 41 \times 47 \times 59 \times 71 \approx 8 \times 10^{53}$$

والتي درسها Robert Griess عام 1982[2]، ومع هذا توصف بأنها بسيطة ومنتهية. والملاحظ أن تصنيف الزمر المنتهية معقد جداً والدراسات تتمحور حول إيجاد عدد الزمر التي لا تتجاوز مرتبتها 2000، وحتى الآن وجد منها كثير، في حين أن أنصاف الزمر لم تحظَ بمثل هذه الحظوة لوجود أعداد كبيرة منها مع ازدياد مرتبها.

فيما يأتي نقدم جدولاً يقارن بين أعداد الزمر وأعداد أنصاف الزمر من المراتب 1 حتى 10.

Table 1

Order N	**Number of Groups (Up to isomorphism)**	**Number of Semigroups (Up to equivalence)**	
1	1	1	
2	1	4	
3	1	18	
4	2	126	[Forsythe **1955**][3]
5	1	1,160	[Motzkin, Selfridge **1955**][4]
6	2	15,973	[Plemmons **1966**][5]
7	1	836,021	[Jürgensen, Wick **1976**][6]
8	5	1,843,120,128	[Satoh, Yama, Tokizawa **1994**][7]
9	2	52,989,400,714,478	[Distler **2009**][8]
10	2	12,418,001,077,381,302,684	[Distler, Jefferson, Kelsey, Kotthoff **2012**][9]

ونلاحظ أنه ما يزال عدد أنصاف الزمر من أجل مراتب أعلى من 10 مجاهيل.

نحاول في عملنا هذا تقديم أسلوب جديد لتصنيف أنصاف الزمر بالاعتماد على البرمجة غرضية التوجه ولغة البرمجة Java، ونعرض النتائج التي توصلنا إليها في تصنيف أنصاف الزمر من المرتبة الثالثة ومن المرتبة الرابعة.

3. تعاريف [10]

● تعرَّف **نصف الزمرة** (semigroup) بأنَّها مجموعة غير خالية S مزودة بقانون تشكيل داخلي وتجميعي، أي إنها تحقق الشرطين:

$$\forall\, x,y,z \in S : (xy)z = x(yz) \text{ و } \forall\, x,y \in S : \; xy \in S$$

● لتكن كل من S,T نصف زمرة، يُقال عن تطبيق مثل $f : S \to T$ إنه **تشاكل** (**أنصاف زمر**) (**(semigroup) homomorphism**) إذا كان يحقق الشرط الآتي:

$$f(xy) = f(x)f(y) \qquad \forall x,y \in S$$

وإذا كان هذا التشاكل تقابلاً يُدعى **تماثلاً** (**isomorphism**) ونكتب $S \cong T$، و يُقال حينها: إن نصفي الزمرتين S,T **متماثلتان** (**isomorphic**).

أمَّا التقابل بين نصفي زمرتين S, T مثل $f : S \to T$ الذي يحقق الشرط:

$$f(xy) = f(y)f(x) \qquad \forall x, y \in S$$

فيُدعى **تماثلاً تخالفياً** (**anti-isomorphism**) بين S, T، و يُقال حينها: إن نصفي الزمرتين S, T **متماثلتان تخالفياً** (**anti-isomorphic**).

ويُقال عن نصفي زمرتين S, T إنهما **متكافئتان** (**equivalent**) إذا كانتا متماثلتين أو متماثلتين تخالفياً.

4. دراسة المجموعة من المرتبة n كنصف زمرة

يُقصد بالمجموعة من المرتبة n أيّ مجموعة تحوي n عنصراً متمايزاً، قد يتبادر إلى الذهن السؤال: هل توجد، من أجل أية مرتبة $n > 1$، نصف زمرة ليست بزمرة ؟ للإجابة عن هذا السؤال نذكر التمهيدية الآتية.

4.1. تمهيدية [10]

توجد نصف زمرة من أي مرتبة $n > 1$، بحيث لا تكون زمرة.

البرهان:

لتكن S مجموعة ما بحيث $|S| > 1$، نستطيع أن نعرف قانون تشكيل داخلي على S كما يأتي: $st = s$، وذلك من أجل كل $s, t \in S$. فنجد بسهولة أن هذا القانون يُعرِّف نصف زمرة على S وذلك بملاحظة أن $(xy)z = xz = x$ و $x(yz) = xy = x$، وذلك من أجل أي $x, y, z \in S$. تُدعى نصف الزمرة هذه نصف زمرة صفرية يسارية.

زد على ذلك، أنَّ نصف الزمرة هذه ليست بزمرة؛ وذلك لأنها لا يمكن أن تملك عنصراً حيادياً. لأننا لو افترضنا جدلاً أنها تملك عنصراً حيادياً عندها ستكون عناصر S كلّها مساويةً لهذا العنصر الحيادي، وهذا سيقودنا لتناقض مع كون $|S| > 1$. ■

ومن المعلوم أن المجموعة وحيدة العنصر $\{a\}$ تُعرِّف نصف زمرة، وذلك إذا عرفنا قانون التشكيل الداخلي عليها بالشكل $aa = a$. وعليه فإن أي مجموعة غير خالية من أي مرتبة يمكن جعلها تُعرِّف نصف زمرة.

يمكن دراسة أي نصف زمرة من المرتبة n بدراسة المجموعة $[n] = \{1,2,\dots,n\}$ كنصف زمرة، وهذا ليس إلا نتيجة مباشرة للمبرهنة الآتية.

4.2. مبرهنة

يوجد تماثل بين أي نصف زمرة منتهية S من المرتبة n وبين $[n] = \{1,2,\dots,n\}$ كنصف زمرة مزودة بقانون تشكيل داخلي (متوافق مع طريقة ترتيب عناصر S).

<u>البرهان:</u>

لتكن $(S,\cdot)$ نصف زمرة منتهية من المرتبة n، عندئذٍ يمكن كتابة S على شكل قائمة بعناصرها المختلفة كما يأتي: $S = \{x_1,x_2,\dots,x_n\}$.

ويمكن أن نعرف التطبيق:

$$\begin{aligned} index : S &\to [n] \\ x_i &\mapsto i \end{aligned}$$

فنجد بسهولة أنه تطبيق تقابل، وتقابله العكسي هو التطبيق:

$$\begin{aligned} index^{-1} : [n] &\to S \\ i &\mapsto x_i \end{aligned}$$

نُعرِّف على $[n]$ قانون تشكيل داخلي $*$ كما يأتي:

$$\begin{aligned} * : [n] \times [n] &\longrightarrow [n] \\ (i,j) &\mapsto i * j \end{aligned}$$

إذْ:

$$i * j = index\left(index^{-1}(i) \cdot index^{-1}(j)\right) = index\left(x_i \cdot x_j\right)$$

فنجد أن $*$ تجعل من $[n]$ نصف زمرة بسبب تحقق الشرطين:

1- شرط الإغلاق:

من أجل أي $i,j \in [n]$ نجد أن:

$$\begin{aligned} i * j &= index\left(index^{-1}(i) \cdot index^{-1}(j)\right) \\ &= index(x_i \cdot x_j) = index(x_k) = k \end{aligned}$$

إذْ x_k هو ناتج $x_i \cdot x_j$ وهو عنصر من S وذلك كون S نصف زمرة فهي مغلقة بالنسبة إلى قانون التشكيل (·)، وعليه فإن الدليل k لابُدَّ أن يكون أحد عناصر المجموعة $\{1,2,\dots,n\}$ ومنه $i * j = k \in [n]$ ، أي إنَّ $[n]$ مغلقة بالنسبة إلى القانون (*) .

2- شرط التجميعية:

من أجل أي $i,j,k \in [n]$ نجد أن:

$$i * (j * k) = index\left(index^{-1}(i) \cdot index^{-1}(j * k)\right)$$
$$= index\left(x_i \cdot index^{-1}\left(index\left(index^{-1}(j) \cdot index^{-1}(k)\right)\right)\right)$$
$$= index\left(x_i \cdot index^{-1}\left(index(x_j \cdot x_k)\right)\right)$$
$$= index\left(x_i \cdot (x_j \cdot x_k)\right)$$

وبشكل مشابه نجد أن:

$$(i * j) * k = index\left((x_i \cdot x_j) \cdot x_k\right)$$

وكون S تحقق الخاصة التجميعية فإن $x_i \cdot (x_j \cdot x_k) = (x_i \cdot x_j) \cdot x_k$

ومنه فإن:

$$index\left(x_i \cdot (x_j \cdot x_k)\right) = index\left((x_i \cdot x_j) \cdot x_k\right)$$

الذي يعني أن $i * (j * k) = (i * j) * k$، أي إنَّ (*) تجميعية.

وعليه فإن $f = index$ هو تطبيق تقابل بين نصفي زمرتين $S, [n]$

$$f : S \to [n]$$
$$x_i \mapsto i$$

ولإثبات التماثل بين نصفي الزمرتين يتبقى لدينا إثبات شرط التشاكل:

من أجل أي $x_i, x_j \in S$ لنثبت أن $f(x_i \cdot x_j) = f(x_i) * f(x_j)$

- $f(x_i \cdot x_j) = index(x_i \cdot x_j)$

- $$f(x_i) * f(x_j) = index(x_i) * index(x_j) = i * j$$
$$= index(index^{-1}(i) \cdot index^{-1}(j))$$
$$= index(x_i \cdot x_j)$$

ومنه ينتج أن f هو تماثل بين نصفي الزمرتين S و $[n]$. ■

5. دراسة الخاصة التجميعية

إنَّ لمن المؤسف عدم قدرتنا على تحديد هل كان جدول كايلي يحقق الخاصة التجميعية بمجرد إلقاء نظرة إليه، على نقيض الخاصة التبديلية، وإن هذا يُعزى إلى كـــون شـــرط التجميعية يعتمد على عبارة بثلاثة متغيرات $(xy)z = x(yz)$ ، في حين أن جـــدول كايلي يظهر نواتج التشكيل لأي متغيرين فقط .[11]

إن هذا يقودنا للبحث عن طريقة للتحقق من الخاصة التجميعية لجدول كايلي.

5.1. طريقة البحث الشامل [Brute-Force method]

لاختبار الخاصة التجميعية على جدول كايلي مُعطى:

Table 2

	1	**2**	...	***n***
1	a_{11}	a_{12}	...	a_{1n}
2	a_{21}	a_{22}	...	a_{2n}
...	...	...	...	...
n	a_{n1}	a_{n2}	...	a_{nn}

نقوم بكتابة قائمة الكلمات كلّها ذات الطول 3 المأخوذة من الأبجدية S، أي من الشكل abc إذْ $a, b, c \in \{1, 2, \dots, n\}$. هذه القائمة ستتضمن n^3 كلمة لأن كـــلاً مـــن a, b, c يمكن اختياره بـــ n إمكانية، ومن ثمَّ سيكون عدد الكلمات الممكن تشكيلها هـــو $n \times n \times n = n^3$ كلمة في هذه القائمة.

نختبر المساواة $(ab)c = a(bc)$ وذلك من أجل كل كلمة abc فـــي القائمـــة السابقة، إذْ نعتمد على جدول كايلي المعطى في حساب كل من الطرفين. فإذا كان جدول

كايلي المُعطى يحقق المساواة من أجل كل كلمة من الكلمات في القائمة عندها فـــإن هـــذا الجدول سيعرِّف نصف زمرة على S. أما إذا كانت المساواة غير محققة من أجل كلمة ما من كلمات القائمة فإن جدول كايلي لا يُعرِّف نصف زمرة على S، وما مـــن داعٍ حينهـــا لاختبار بقية الكلمات التي تلي تلك الكلمة في القائمة.

وعليه من خلال هذه الطريقة سنقوم بـــn^3 خطوة على الأكثر للتحقق: هل كان جدول كايلي مُعطى يُعرِّف نصف زمرة أم لا؟

فيما يأتي نُقدِّم صياغة لهذه الطريقة بشكل خوارزمية تأخذ دخلاً لها مصفوفة A مـــن البعد $n \times n$ تمثل جدول كايلي، وتعيد قيمة منطقية true إذا كان جدول كايلي يُعـــرِّف نصف زمرة أو false إذا كان لا يُعرِّف نصف زمرة.

Algorithm 1-check a multiplication table represented by a matrix A defined on the set $S=[n]=\{1,2,\ldots,n\}$ if it defines a semigroup or not using Brute-Force method

Input: a matrix A that represents the multiplication table defined on S

Output: true if A defines a semigroup, false if it doesn't

$t \leftarrow$ true (t is a boolean variable refers to the statues of the multiplication table whether it defines a semigroup or not)

Outer **:**
for each $a \in \{1,2,\ldots,n\}$ **do**
  for each $b \in \{1,2,\ldots,n\}$ **do**
    for each $c \in \{1,2,\ldots,n\}$ **do**
      if not $(T(A,a,b,c))$ **then** (T is a function to test association property)
        $t \leftarrow$ false
        break Outer
      end if
    end for
  end for
end for
return t

Function $T(A,i,j,k)$ $(i,j,k \in S)$
  Return $A[A[i][j]][k] = A[i][A[j][k]]$ (return $(ij)k = i(jk)$)

وتطبيقاً لهذه الخوارزمية قمنا بكتابة برنامجٍ بلغة البرمجة Java بحيث يقرأ هذا البرنامج جدول كايلي متمثلاً بمصفوفة مربعة A، ثم يستخدم الخوارزمية أعلاه ليحدد هل كانت A تُعرِّف على $[n]$ نصف زمرة أم لا؟

6. توليد أنصاف الزمر

استُخدمتْ كذلك الخوارزمية أعلاه في كتابة برنامج بلغة البرمجة Java يبحث بين جداول كايلي جميعها التي يمكن أن نعرفها على المجموعة $[n]$، والبالغ عددها n^{n^2} ، ونتج لدينا أنصاف الزمر جميعها التي يمكن تعريفها، وحُفظتْ النتائج بصيغة ملفات مضغوطة gzip [12] ، وذلك من أجل حالتي $n = 3$ و $n = 4$.

استُخدمتْ تقنيات البرمجة غرضية التوجه بتعريف صف يمثل النوع المجرد "نصف زمرة"، وذلك لتوفير مرونة أكبر في التعامل مع أنصاف الزمر الناتجة.

بدايةً زودنا كل كائن من الصف "نصف زمرة" الذي يمثل نصف زمرة، بالعناصر الآتية:

- M: مصفوفة تمثِّل جدول كايلي الذي يصف قانون التشكيل الداخلي المعرف على نصف الزمرة.
- dim: مرتبة نصف الزمرة أو بعد مصفوفة جدول كايلي.

عَرَّفنا متجهتين من كائنات الصف "نصف زمرة"، Semigroups3 وSemigroups4 لنخزن فيها أنصاف الزمر من المرتبة الثالثة ومن المرتبة الرابعة على الترتيب. وحفظناهما بملفين Semigroups3.gz وSemigroups4.gz .

استغرق برنامجنا ما لا يزيد على 2 ثانية في توليد أنصاف الزمر من المرتبة الثالثة، في حين استغرق ما يقارب 44 دقيقة في توليد أنصاف الزمر من المرتبة الرابعة، وذلك باستخدام حاسوب شخصي ذي معالج Core i3 2.10GHz وذاكرة 4GB، ولذلك حُفظت النتائج بصيغة ملفات مضغوطة gzip بغية تحميلها بسهولة للذاكرة وعدم إعادة توليدها مرة أخرى عند العمل على معالجتها، ومرحلة تحميل أنصاف الزمر للذاكرة لم تستغرق ثانية واحدة في كلتا الحالتين $n = 3$ و $n = 4$.

7. تصنيف أنصاف الزمر

كمرحلة ثانية بعد أن ولّدنا أنصاف الزمر من المرتبة الثالثة ومن المرتبة الرابعة، وحفظنا المخرجات، نقوم بتحميل الملفات التي قمنا بحفظها مسبقاً للذاكرة من أجل العمل على تصنيفها.

لكن قبل البدء بمرحلة التصنيف نذكر التمهيدية المساعدة الآتية التي ساعدتنا في عملنا لاحقاً.

7.1. تمهيدية مساعدة

لتكن $(S,\cdot),(S`,*)$ نصفي زمرتين منتهيتين، وليكن $f:S\longrightarrow S`$ تماثلاً بينهما.

ولنستخدم الترميز $f(x)=x` \quad \forall x\in S$، عندئذٍ :

1. إذا ملكت S عنصراً صفرياً فإن $S`$ ستملك عنصراً صفرياً كذلك.
2. إذا ملكت S عنصراً حيادياً فإن $S`$ ستملك عنصراً حيادياً كذلك.
3. إذا كان $\#\{s_1\cdot s_2\ ;\ s_1,s_2\in S\}=m$ فإن $\#\{s`_1 * s`_2\ ;\ s`_1,s`_2\in S`\}=m$

(إذْ يُقصد بالرمز X# عدد عناصر المجموعة X)

<u>البرهان:</u>

1- إذا ملكت S عنصراً صفرياً وليكن 0، فإنه من أجل كل $s\in S$ يكون

$$s\cdot 0=0\cdot s=0$$

ولوجود التماثل f فإنه من أجل كل $s`\in S`$ يوجد عنصر وحيد $s\in S$ بحيث $s`=f(s)$ ، وعليه فإنه من أجل كل $s`\in S`$ يمكن أن نكتب:

$$s`*f(0)=f(s)*f(0)=f(s\cdot 0)=f(0)$$
$$f(0)*s`=f(0)*f(s)=f(0\cdot s)=f(0)$$

الذي يعني وجود عنصر $0`=f(0)$ يحقق $s`*0`=0`*s`=0`$ من أجل كل $s`\in S`$

أي إنَّ $S`$ تملك عنصراً صفرياً كذلك.

2- إذا ملكت S عنصراً حيادياً وليكن 1 ، فإنه من أجل كل $s\in S$ يكون

$$s \cdot 1 = 1 \cdot s = s$$

ولوجود التماثل f فإنه من أجل كل $s` \in S`$ يوجد عنصر وحيد $s \in S$ بحيث $s` = f(s)$ ، وعليه فإنه من أجل كل $s` \in S`$ يمكن أن نكتب :

$$s` * f(1) = f(s) * f(1) = f(s \cdot 1) = f(s) = s`$$

$$f(1) * s` = f(1) * f(s) = f(1 \cdot s) = f(s) = s`$$

الذي يعني وجود عنصر $1` = f(1)$ يحقق $s` * 1` = 1` * s` = s`$ من أجل كل $s` \in S`$

أي إنَّ $S`$ تملك عنصراً حيادياً كذلك.

3- إذا كان $\#\{s_1 \cdot s_2 \ ; \ s_1, s_2 \in S\} = m$ عندئذٍ سيكون هناك m إمكانية لـ $s_1 \cdot s_2$.

ومن أجل كل $s`_1, s`_2 \in S`$ يوجد عنصران وحيدان $s_1, s_2 \in S$ بحيث

$$s`_1 = f(s_1), \qquad s`_2 = f(s_2)$$

وعليه فإن $s`_1 * s`_2 = f(s_1) * f(s_2) = f(s_1 \cdot s_2)$.

وكون $s_1 \cdot s_2$ يملك m إمكانية، فإن ذلك يقتضي أن $f(s_1 \cdot s_2)$ يملك m إمكانية كذلك، أي إنَّ $s`_1 * s`_2$ يملك m إمكانية وعليه فإن $\#\{s`_1 * s`_2 \ ; \ s`_1, s`_2 \in S`\} = m$. ■

وفقاً لما ذكرناه أعلاه صُنِّفت أنصاف الزمر من المرتبة n التي وُلِّدت مسبقاً ضمن الأنواع الآتية:

- أنصاف الزمر المتماثلة تخالفياً
- أنصاف الزمر التي تملك عنصراً صفرياً وعنصراً حيادياً
- أنصاف الزمر التي تملك عنصراً حيادياً ولا تملك عنصراً صفرياً
- أنصاف الزمر التي تملك عنصراً صفرياً ولا تملك عنصراً حيادياً
- أنصاف الزمر التي لا تملك عنصراً صفرياً ولا عنصراً حيادياً (مجموعة قيم جدول كايلي لها تتألف من عنصرين)
- أنصاف الزمر التي لا تملك عنصراً صفرياً ولا عنصراً حيادياً (مجموعة قيم جدول كايلي لها تتألف من ثلاثة عناصر)

- أنصاف الزمر التي لا تملك عنصراً صفرياً ولا عنصراً حيادياً (مجموعة قيم جدول كايلي لها تتألف من n عنصراً)

بملاحظة أنَّ من غير الممكن وجود أي نصف زمرة لا تملك عنصراً صفرياً ولا عنصراً حيادياً ومجموعة قيم جدول كايلي لها تتألف من عنصر واحد، لأن نصف الزمرة التي مجموعة قيم جدول كايلي لها تتألف من عنصر واحد هي نصف الزمرة الصفرية (the null semigroup) التي تملك عنصراً صفرياً أكثر وضوحاً.

لإجراء هذا التصنيف المقترح، أضفنا الدوال التالية إلى الصف "نصف زمرة":

- boolean has_zero();
- boolean has_identity();
- int number_of_elements_in_Matrix();

نورد فيما يأتي الخوارزميات التي كتبناها لاختبار هل كان نصف زمرة S يملك عنصراً صفرياً؟ وهل كان نصف زمرة S يملك عنصراً حيادياً؟ لحساب عدد عناصر مجموعة قيم جدول كايلي لنصف زمرة S.

Algorithm 2 - has_zero( ) function Determine whether a semigroup $(S,*)$ has a zero or not
Input: a semigroup S
Output: true if S has a zero element, false if it doesn't have

for each $i \in S$ **do**
    $counter \leftarrow 0$
    for each $j \in S$ **do**
        if $((i * j = i) \text{ and } (j * i = i))$ **then**
            $counter \leftarrow counter + 1$
        end if
    end for
    if $(counter = \|S\|)$ **then**
        return $true$
    end if
end for
return $false$

Algorithm 3 - has_identity ( ) function Determine whether a semigroup $(S,*)$ has an identity or not

Input: a semigroup S

Output: true if S has an identity element, false if it doesn't have

```
for each i ∈ S do
    counter ← 0
    for each j ∈ S do
        if ((i * j = j) and (j * i = j)) then
            counter ← counter + 1
        end if
    end for
    if (counter = |S|) then
        return true
    end if
end for
return false
```

Algorithm 4-number_of_elements_in_Matrix ( ) function Determine the number of elements in the set of the multiplication table values of a semigroup S

Input: a semigroup S

Output: an integer value represents the number of elements in the set of the multiplication table values of S

```
counter ← 0
BooleanS ← [ ]  ( an array of boolean values of length |S| )
for each i ∈ S do
    for each j ∈ S do
        BooleanS[ i * j ] = true
    end for
end for
for k ∈ {1,2,…,|S|}
    if ( BooleanS[ k ] = true ) then
        counter ← counter + 1
    end if
end for
return counter
```

لكي نحدد نوع كل نصف زمــرة فـــي Semigroups3 وSemigroups4 اســتخدمنا متجهتين للأدلة من الأعداد الصحيحة index3 وindex4 لهما طــول المتجهــات نفــسه Semigroups3 وSemigroups4 على الترتيب، وتوافق القيمة الصحيحة الموجودة فـــي الموضع i في المتجهة index3 (index4) نصف الزمرة في الموضـــع i فـــي المتجهـــة Semigroups3 (Semigroups4)، إذْ يشير كل عدد صحيح إلى نوع معين:

- -1 نوع غير محدد
- 0 أنصاف الزمر المتماثلة تخالفياً
- 1 أنصاف الزمر التي تملك عنصراً صفرياً وعنصراً حيادياً
- 2 أنصاف الزمر التي تملك عنصراً حيادياً ولا تملك عنصراً صفرياً
- 3 أنصاف الزمر التي تملك عنصراً صفرياً ولا تملك عنصراً حيادياً
- 4 أنصاف الزمر التي لا تملك عنصراً صفرياً ولا عنصراً حيادياً (مجموعة قيم جدول كايلي لها تتألف من عنصرين)
- 5 أنصاف الزمر التي لا تملك عنصراً صفرياً ولا عنصراً حيادياً (مجموعة قيم جدول كايلي لها تتألف من ثلاثة عناصر)
- 6 أنصاف الزمر التي لا تملك عنصراً صفرياً ولا عنصراً حيادياً (مجموعة قيم جدول كايلي لها تتألف من أربعة عناصر)

ابتداءً بالقيمة -1 في المواضع جميعها في المتجهتين index3 وindex4، نحـــدد أولاً أنصاف الزمر المتماثلة تخالفياً.

ولكي نحدد أنصاف الزمر المتماثلة تخالفياً احتجنا لطريقةٍ نختبر فيها: هل كان نـــصفا زمرتين متماثلتين، متماثلتين تخالفياً. لذلك قمنا بكتابة الخوارزميتين الآتيتين:

Algorithm 5 - **checking isomorphic**
check if two semigroups, whose multiplication tables represented by two matrices A, B, defined on the set $[n] = \{1,2,\dots,n\}$, are isomorphic or not

Input: two matrices A, B that represent the multiplication tables of the two semigroups that defined on $[n]$

Output: true if the two semigroups are isomorphic, false if they aren't

1: **for each** $f_k \in S_n$ **do** (where S_n is the symmetric group)
2: Outer **:**
3: **for each** $i \in \{1,2,\dots,n\}$ **do**
4: **for each** $j \in \{1,2,\dots,n\}$ **do**
5: **if** $f_k(A[i][j]) = B[f_k(i)][f_k(j)]$ **then**
$(f_k(i * j) = f_k(i) \cdot f_k(j))$
where $*$ the operation defined on A,
$\cdot$ the operation defined on B
6: $s \leftarrow 0$
7: **break** Outer
8: **else**
9: $s \leftarrow s + 1$
10: **end if**
11: **end for**
12: **end for**
13: **if** $(s = n^2)$ **then** (n^2 is the number of equalities of the form $f_k(i * j) = f_k(i) \cdot f_k(j)$ that f_k should verify in order to satisfy the homomorphism condition)
14: return $true$
15: **end if**
16: **end for**
17: return $false$

الخوارزمية الأخرى التي تختبر هل كان نصفا زمرتين متماثلتين تخالفياً تنتج من الخوارزمية السابقة باستبدال التعليمة الشرطية في السطر 5 بالتعليمة الشرطية الآتية:

$$\textbf{if } f_k(A[i][j]) = B[f_k(j)][f_k(i)] \textbf{ then}$$

الآن لتحديد أنصاف الزمر المتماثلة تخالفياً نمر بالعناصر المتجهة index3 (index4) ونتحقق: هل كانت القيمة التي نمر بها i_0 مساوية لـــ-1 أم لا، فــإذا كانــت مــساوية لـــ-1 نمر بالأدلة اللاحقة لها كلّها، فإذا كانت نصف الزمرة اللاحقة الموافقة لدليل ما j_0 متماثلة تخالفياً مع نصف الزمرة الموافقة للدليل i_0 وغير متماثلة معها، فإننا نسند القيمة 0 إلى المتجهة index3 (index4) في الموضع الموافق لـ j_0 .

وبعد تطبيق هذه الخطوة السابقة بعناصر كل من index3 وindex4 فـإن قـيم هـذه المتجهات ستكون إما 1- إذ لم نحدد نوع أنصاف الزمر الموافقة بعد، وإما 0 من أجـل أنصاف الزمر المتماثلة تخالفياً مع أحد أنصاف زمر الأخرى من النوع غيـر المحـدد. بعبارة أخرى، نكون قد عزلنا أنصاف الزمر المتماثلة تخالفياً، بحيث نستطيع البحث عـن صفوف أنصاف الزمر المتكافئة في بقية أنصاف الزمر.

الآن نمر مرة أخرى بعناصر (index4) index3 بحثاً عن النوع غير المحـدد بعـد، ونستخدم الدوال( )has_zero و( ) has_identity و number_of_elements_in_Matrix ( ) التي عرّفناها مسبقاً لتحديد نوع كـل مـن أنـصاف الزمـر فـي Semigroups3 (Semigroups4) وفقاً للقيم التي تعيدها هذه الدوال.

بعد هذه الخطوات، إذا فحصنا المتجهتين index3 وindex4 فـسنجد عـدد أنـصاف الزمر من كل نوع ونلاحظ أن كل نصف زمرة له أحد الأنواع التـي عرفناهـا مـسبقاً، ونلاحظ كذلك عدم وجود أي أنصاف زمر متبقية من النوع غير المحدد.

إن البرنامج الذي كتبناه بلغة البرمجة Java. الذي يقوم بالخطوات المشروحة أعـلاه، أعطانا النتائج الآتية التي نقدمها في الجدولين Table 3 وTable 4.

Semigroups3

Anti-Isomorphic & not Anti-Isomorphic Semigroups	
Non-Determined semigroups	0
Anti-Isomorphic semigroups	25
Not Anti-Isomorphic semigroups	88
Total number of semigroups	113

The number of semigroups of each type	
Anti-Isomorphic semigroups	25
Semigroups with both zero and identity	18
Semigroups with identity but without zero	12
Semigroups with zero but without identity	33
Semigroups without zero & without identity (two elements in the multiplication table)	18
Semigroups without zero & without identity (three elements in the multiplication table)	7
Total number of semigroups	113

Table 3

Semigroups4

Anti-Isomorphic & not Anti-Isomorphic Semigroups	
Non-Determined semigroups	0
Anti-Isomorphic semigroups	1113
Not Anti-Isomorphic semigroups	2379
Total number of semigroups	3492

The number of semigroups of each type	
Anti-Isomorphic semigroups	1113
Semigroups with both zero and identity	264
Semigroups with identity but without zero	236
Semigroups with zero but without identity	1068
Semigroups without zero & without identity (two elements in the multiplication table)	72
Semigroups without zero & without identity (three elements in the multiplication table)	324
Semigroups without zero & without identity (four elements in the multiplication table)	415
Total number of semigroups	3492

Table 4

ولكي نكمل هذه الرحلة الاستكشافية نحتاج أن نحدد صفوف أنصاف الزمر المتماثلــة لكل نوع من الأنواع التي أوجدناها.

من أجل ذلك قمنا بتعريف متجهات من أنصاف الزمر كل منها تمثل نوعاً من الأنواع التي أوجدناها، ومن أجل كل متجهة بحثنا عن صفوف أنصاف الزمر المتماثلة فيها.

كتبنا خوارزمية لدالة search_for_iso تأخذ كدخل لها متجهة من أنــصاف الزمــر، وتبحث عن صفوف أنصاف الزمر المتماثلة فيها، وتعيد متجهة من الأدلة الصحيحة لهــا طول متجهة الدخل نفسه، تمثل كل قيمة صحيحة فيها صفاً من أنصاف الزمر المتماثلة.

نعرض فيما يأتي الخوارزمية search_for_iso التي كتبناها

Algorithm 6 - search_for_iso
search for isomorphic semigroups in an array of semigroups

Input: an array X of semigroups

Output: an array of indices that represent the isomorphic semigroups where each number represent one class

$Index \leftarrow [-1,-1,\dots,-1]$ (an array of integer values of length $|X|$)
n_class $\leftarrow 0$
for each $i \in \{1,2,\dots,|X|\}$ **do**
    if $Index[i] < 0$ **then**
        n_class $\leftarrow$ n_class +1
        $index[\,i\,] \leftarrow$ n_class
        for each $j \in \{i+1, i+2,\dots,|X|\}$ **do**
            if ($X\,[\,i\,] \cong X\,[\,j\,]$) **then**
            (we use checking isomorphic function that we'd defined)
                $Index\,[\,j\,] \leftarrow$ n_class
            end if
        end for
    end if
end for
return $Index$

وباستدعاء هذه الدالة search_for_iso على كل متجهة من متجهات أنصاف الزمــر من كل نوع نستنتج عدد الصفوف في كل نوع.

تعرض الجداول الآتية table 5 وtable 6 وtable 7، النتائج والأرقام التـــي توصـــلنا إليها في نهاية عملنا.

Table 5

Semigroups3		
Type	**#semigroups**	**#classes**
Semigroups3_AntiIsomorphic	25	6
Semigroups3_with_zero_with_identity	18	3
Semigroups3_with_identity_without_zero	12	3
Semigroups3_with_zero_without_identity	33	7
Semigroups3_without_zero_without_identity_2	18	3
Semigroups3_without_zero_without_identity_3	7	2
Total (including the anti-semigroups)	113	24
Total (excluding the anti-semigroups)	88	18

Table 6

Semigroups4		
Type	**semigroups#**	**classes#**
Semigroups4_AntiIsomorphic	1113	62
Semigroups4_with_zero_with_identity	264	13
Semigroups4_with_identity_without_zero	236	14
Semigroups4_with_zero_without_identity	1068	54
Semigroups4_without_zero_without_identity_2	72	5
Semigroups4_without_zero_without_identity_3	324	15
Semigroups4_without_zero_without_identity_4	415	25
Total (including the anti-semigroups)	3492	188
Total (excluding the anti-semigroups)	2379	126

Table 7

Semigroups3	**Semigroups4**
Total semigroups: 113 semigroups. Total Types: 6 types. Total Classes (up to isomorphism): 24 classes. Total Classes (up to equivalence): 18 classes.	Total semigroups: 3492 semigroups. Total Types: 7 types. Total Classes (up to isomorphism): 188 classes. Total Classes (up to equivalence): 126 classes

كما يمكن الاطلاع على القائمة الكاملة من أنصاف الزمر من كل نوع التي أخرجهــا برنامجنا من خلال استعراض الملفين الآتيين أو تحميلهما.

Semigroups3.pdf [http://goo.gl/P36YhY]

Semigroups4.pdf [http://goo.gl/QGSuYM]

REFERENCES

[1] Clifford, A. H. and Preston, G. B. (1961). The Algebraic Theory of Semigroups, Vol. I. Mathematical Surveys of the American Mathematical Society No.7. p.1.

[2] Griess, R. L. (1982). The friendly giant. Inventiones Mathematicae 69 (1): 1-102

[3] Forsythe, G. E. (1955). SWAC computes 126 distinct Semigroups of order 4. Proc. Amer. Math. Soc., 6:443-447.

[4] Motzkin, T. S. and Selfridge, J. L. (1956). Semigroups of order five. presented in The November meeting in Los Angeles.

[5] Plemmons, R. J. (1967). There are 15973 Semigroups of order 6. Math. Algorithms, 2:2–17.

[6] Jürgensen, H. (1977/78). Computers in Semigroups. Semigroup Forum, 15(1):1–20.

[7] Satoh, S., Yama, K., and Tokizawa, M. (1994). Semigroups of order 8. Semigroup Forum, 49(1):7–29, 1994.

[8] Distler, A., Kelsey, T., and Mitchell, J. (2009). Enumeration of Semigroups of Order 9. CIRCA Talk - June 2009

[9] Distler, A., Jefferson, C., Kelsey, T., and Kotthoff, L. (2012). The Semigroups of Order 10. Principles and Practice of Constraint Programming Lecture Notes in Computer Science, p. 883-899.

[10] Mitchell, J. (2010). Lecture Notes on Semigroup Theory. The School of Mathematics and Statistics at the University of St Andrews. Lecture Notes 1, 3.

[11] Cayley table - Wikipedia [http://en.wikipedia.org/wiki/Cayley_table] Access Date: 2/6/2013

[12] RFC 1952 - GZIP file format specification version 4.3 [https://tools.ietf.org/html/rfc1952] - Access Date: 26/7/2013