النظام الخبير لبعض الإختبارات المتعددة اللامعلمية

أ.م.د. فارس غانم أحمد* م.د. ريم علي عبد الحسين الجراح**

المستخلص:

كثيراً ما نلجأ إلى الاختبارات اللامعلمية لمعرفة وجود فروقات معنوية من عدمها بين أكثر من عينتين، وباستخدام أي من البرامج الجاهزة والمتوفرة في الحقائب البرمجية. إلاّ أن مسؤولية الفروقات المعنوية الناتجة بين أيّ من العينات المستخدمة يبقى مبهماً ويحتاج إلى توضيح عندما تكون المقارنة بين أكثر من عينتين، وبالتالي يتوجب معرفة أي من تلك العينات مختلفة معنوياً وأي منها ليست كذلك.

وفي هذا البحث تم تقديم أهم الاختبارات اللامعلمية ذات العلاقة بأكثر من عينتين، وهما: اختبار كروسكال ويلز لتحليل التباين للرتب واختبار فريدمان لتحليل التباين من الدرجة الثانية للرتب، كما تم تسليط الضوء على المقارنات المتعددة اللازم إجرائها بعد الاختبارين المذكورين حول مسألة ما قيد الاختبار، ليس فقط إحصائياً وإنما برمجياً. إذ صمم نظاماً خبيراً وبالاعتماد على تقنية الحزمة البرمجية (Macro) والموجودة في النظام الـجاهز (Minitab).

ولاختبار كفاءة النظام المعد، تم تطبيقه على عدة أمثلة، وتم اختيار المثالين المقتبسين من (القرشي، ص200 و ص213)[1] " لكل اختبار "، وتم عرض نتائج تطبيقهما في النظام الخبير النتائج التي تم التوصل إليها وبشكل واضحة.

Abstract:

Mightily we refuge (resort) to the Non-Parametric test to success output about some problems, but this case still abstruse and need to illustration when the comparison between more than two samples, so behoove knowing which of these samples are significant different and which of them are not.

In this paper presentation to superior Non-Parametric test witch dealing with more than two samples, they are: The Kruskal-Wales One-Way Analysis of Variance by Rank & the Friedman Two-Way Analysis of Variance by Rank, and also to shed the light to multiple compare must be dune after testing, that is not only statistically but programming by depend on Macro in MINITAB System.

المقدمة:

إن المُسلّمات التي تعتبر أساساً في الاختبارات المعلمية، ومنها: "التوزيع الطبيعي"، "التباين المتجانس"، قد لا تتحقق دائماً في المجتمعات المدروسة، وكثيراً ما نحتاج لوسائل أخرى لعمليات الاستدلال بحيث لا تتطلب تحقق تلك المُسلّمات أو الشروط القاسية. والإحصاء اللامعلمي، هو الذي يوفر لنا هذه الوسيلة، لأنه يصلح في كثير من الظروف التي لا تتطلب إلاّ على مُسلّماتٍ عامة للغاية[2].

* كلية طب الأسنان / جامعة الموصل

** كلية طب الأسنان / جامعة الموصل

(1) القرشي، ص200 و ص213.

[2] آدم، ص 120.

وقد بات معروفاً وجود اختبارات لامعلمية تناظر الاختبارات المعلمية في ظروف استخدامها من حيث تصميم التجربة أو طبيعة المشكلة البحثية[1]. وبالرغم من أن معظم الحقائب البرمجية الإحصائية الجاهزة تفرد للاختبارات اللامعلمية مجالاً مقبولاً للتطبيق المباشر والتوصل إلى النتيجة المطلوبة بسلاسة ومرونة. إلاّ أنه يعاب عليها القصور في بعض الاختبارات عندما نحتاج إلى المقارنات المتعددة التي تبرز أهميتها عندما تكون نتيجة الاختبار رفض عدم وجود اختلاف بين أوساط المجتمعات المبحوثة.

من هنا ظهرت فكرة هذا البحث الذي يهدف إلى تصميم حقيبة برمجية (النظام الخبير) الذي يعمل على إجراء المقارنات المتعددة المهمة بعد تطبيق المختبر الإحصائي المناسب لتصميم التجربة قيد الاهتمام.

وقد تم اختيار المؤشر الإحصائي الخاص باختبار كروسكال- ويلز واختبار فريدمان والمقارنات المتعددة اللازم إجرائها بعد ثبات معنوية الاختلاف بين العينات المبحوثة، لغرض تصميم البرنامج الخبير ليعمل على إجراء المقارنات المتعددة مباشرة للاختبارين المذكورين.

تم التقديم في هذا البحث كل من الجانب النظري الذي يهتم بالتفاصيل النظرية لكل من اختبار كروسكال- ويلز واختبار فريدمان إحصائياً ومن ثم الجانب العملي الذي يهتم بتطبيق الخوارزمية الخاصة بالحقيبة البرمجية على بيانات مثالين تجريبيين، ومناقشة النتائج، ثم تقديم أهم الاستنتاجات والتوصيات.

هدف البحث:

يهدف البحث إلى تصميم حقيبة برمجية (نظام خبير)، لتعطي نتيجة المقارنات المتعددة الواجب تنفيذها بعد كل من اختبار كروسكال- ويلز واختبار فريدمان وحسب صلاحية الاختبار على العينات المبحوثة، باعتبارهما إحدى الاختبارات ذات العلاقة بأكثر من عينتين.

الجانب النظري

1. اختبار كروسكال- ويلز لتحليل التباين للرتب:

The Kruskal -Wallis One Way Analysis of Variance by Rank:

قدم كل من كروسكال و ويلز مختبراً إحصائياً سمي باسمهما يعتمد على الرتب ويناسب تجارب ذات تصميم عشوائي كامل[2]، إذ يفترض هذا الاختبار وجود K من العينات وبأحجام $n_1, n_2, \ldots, n_k$، وأن المشاهدات مستقلة داخل وبين العينات على أن المجتمعات المسحوبة منها العينات متماثلة، وإن البيانات على الأقل رئيسية ومتغيراتها مستمرة. أما فرضيات الاختبار فهي[3]:

H_0: قيمة الوسيط لـ K من المجتمعات متماثلة.

H_1: قيمة الوسيط لمجتمع واحد على الأقل غير متماثل مع البقية.

1-1. خطوات اختبار كروسكال- ويلز:

يعرض كل من[4] (Glantz) و (Siegel) الخطوات المتبعة للحصول على إحصاءة الاختبار الخاصة بكروسكال- ويلز، وكما يأتي:

1. ترتب مشاهدات كل العينات تصاعدياً وكأنها عينة واحدة.
2. يعوض عن قيمة كل مشاهدة بترتيبها الذي أخذته بعد إجراء الخطوة أولاً. مع ملاحظة أن قيم المشاهدات المتساوية يعطى لها نفس الرتبة.
3. حساب معدل الرتب المكررة في حالة وجودها.
4. إعادة توزيع الرتب حسب مشاهدات العينات التابعة لها بحيث نجد لكل عينة بدلاً من قيمة المرتبة التي حصلت عليها.
5. احتساب المؤشر الإحصائي، وفق ما يلي:

أ. في حالة عدم وجود رتب مكررة:

$$KW = \left[\frac{12}{N(N+1)} \sum_{i=1}^{k} \frac{R_i^2}{n_i} \right] - 3(N+1) \quad \ldots\ldots\ldots\ldots\ldots \textbf{(1-1 a)}$$

---

[1] (Petrie., p. 175).

[2] (Steel, R.G.D. & Torrie, J.H., p. 444).

[3] (القرشي، ص 198)

[4] (Glantz, pp. 346-349) و (Siegel, pp. 206-210).

ب. في حالة وجود رتب مكررة:

$$KW = \frac{\left[\frac{12}{N(N+1)}\sum_{i=1}^{k}\frac{R_i^2}{n_i}\right] - 3(N+1)}{1-\sum(t_i^3 - t_i)/(N^3 - N)} \quad \text{..... (1-1 b)}$$

حيث أن:

k	n_i	N	R_i	t
عدد العينات	عدد المشاهدات في العينة i	$\sum_{i=1}^{k} n_i$	مجموع الرتب في العينة i	عدد رتب المشاهدات المكررة في العينة i

6. اتخاذ القرار: فإذا كانت قيمة المؤشر الإحصائي أكبر من قيمتها الجدولية، وكما هو معروف في أدبيات اختبار الفرضيات سترفض فرضية العدم H_0، أي أن قيمة الوسيط لـ K من المجتمعات غير متماثلة. وعليه ينبغي معرفة أي مجتمع من هذه المجتمعات المبحوثة يحمل فروقات عن البقية، وهذا يتم وفق الفقرة الآتية:

2-1. المقارنات المتعددة بعد اختبار كروسكال- ويلز :

من المعلوم أنه في الاختبارات المعلمية عند رفض الفرضية العدمية أي وجود فروقات بين المجتمعات المبحوثة، يجب استخدام ما يسمى (Post Hoc Multiple Comparisons) والموجود في أغلب الأنظمة الإحصائية الحاسوبية الجاهزة لاختيار أحد الاختبارات التي تُظهر أيٍّ من المتوسطات سببت الاختلافات، والتي تدعى الاختلافات المعنوية في تحليل التباين المستخدم، وعلى سبيل المثال وليس الحصر: اختبارات: LSD test، Duncan test، Dunnett test، واختبار Scheffe test وغيرها كثير من الاختبارات التي تبين مصدر الاختلافات في العينات المبحوثة.

وكذلك الحال في الاختبارات اللامعلمية، تثار نفس المشكلة في حالة رفض فرضية العدم التي تنص على تساوي قيمة الوسيط لـ K من المجتمعات: "فما هي الوسائط المختلفة؟"، أو "ما هو الوسيط الذي يختلف عن بقية الوسائط؟".

إن الكثير من الباحثين ناقشوا هذا الموضوع ولعل Dunn - 1964 كان أفضلهم، وقد أوضح كل من[1] (القرشي)، (Glantz) و (Siegel) فكرة المقارنات بالخطوات الآتية:

1. إيجاد الوسيط للرتب لكل عينة و لتكن $\overline{R_i}$.

2. إيجاد قيمة المتباينة أدناه:

أ. في حالة عدم وجود رتب مكررة:

$$\left|\overline{R_i} - \overline{R_j}\right| \le Z_{\alpha/k(k-1)}\sqrt{\frac{N(N+1)}{12}\left(\frac{1}{n_i}+\frac{1}{n_j}\right)} \quad \text{..... (1-2 a)}$$

ب. في حالة وجود رتب مكررة:

$$\left|\overline{R_i} - \overline{R_j}\right| \le z_{\alpha/k(k-1)}\frac{\sqrt{[N(N+1) - \sum(t_i^3 - t_i)]\left(\frac{1}{n_i}+\frac{1}{n_j}\right)}}{12(N-1)} \quad \text{.....(1-2 b)}$$

2. اختبار فريدمان لتحليل التباين من الدرجة الثانية للرتب:

The Friedman Two-Way Analysis of Variance by Ranks:

قدم فريدمان اختباراً يناسب أغلب التجارب ذات تصميم القطاعات العشوائية الكاملة بأكثر من معاملتين[2] ، إذ يفترض هذا الاختبار وجود (b) من العينات " تعامل معاملة القطاعات" وبحجم K. فمثلاً: X_{ij} تمثل المشاهدة j'th في العينة i'th. أي أن المعالجات تمثل الأعمدة والصفوف هي القطاعات. ومهم

[1] (القرشي، ص205)، (Glantz, p.350) و (Siegel, p. 213).
[2] (Steel, R.G.D. & Torrie, J.H., p. 546).

جداً أن يكون متغير المشاهدات مستمر والتفاعلات بين القطاعات والمعالجات معدومة. أما فرضيات الاختبار فهي[1]:

H_0: المعالجات داخل القطاعات متماثلة.

H_1: على الأقل معالجة واحدة تأثيرها أكبر من البقية.

2- 1. خطوات اختبار فريدمان:

يعرض كل من[2] (Glantz) و (Siegel) الخطوات اللازمة للحصول على المؤشر الإحصائي الخاص باختبار فريدمان، وكما يأتي:

1. ترتب البيانات ضمن القطاع الواحد "بشكل مستقل" تنازلياً أو تصاعدياً، ثم يعوض عن القيم الأصلية برتبها، (يستخدم معدل الرتب عند تكرارها).
2. حساب مجموع الرتب R_i لكل عمود.
3. حساب المؤشر الإحصائي لفريدمان، وكالآتي:

في حالة عدم وجود رتب مكرره (2-1 a)...

$$Fr = \left[\frac{12}{bk(k+1)}\sum_{j=1}^{k} R_j^2\right] - 3b(k+1)$$

في حالة وجود رتب مكرره (2-1 b) ...

$$Fr = \frac{\left[\frac{12}{bk(k+1)}\sum_{j=1}^{k} R_j^2\right] - 3b(k+1)}{[1-\sum T_i / bk(k^2-1)]}$$

حيث أن:

T_i	R_i	t_i	b	k
$\sum(t_i^3 - t_i) = T_i$	مجموع الرتب لكل عمود	عدد المشاهدات اللاتي لها نفس الرتبة في i-th	عدد القطاعات	عدد المعالجات

4. اتخاذ القرار: في حالة كون قيمة المؤشر الإحصائي اكبر من قيمتها الجدولية ترفض فرضية العدم، أي أن هناك على الأقل معالجة واحدة ذات تأثير مختلف عن البقية.

2-2. المقارنات المتعددة بعد اختبار فريدمان:

كما سبق في اختبار كروسكال، فعند وجود فروقات معنوية في هذا الاختبار، فينبغي معرفة أي من المعالجات ذات التأثير المختلف عن بقية المعالجات، والتي أدت بالنهاية إلى رفض فرضية العدم، ويتم ذلك من خلال الآتي[3]:

1) إيجاد مجموع رتب كل معالجه (كل عمود) R_i.
2) تحديد مستوى المعنوية لتحديد القيمة الجدولية لـ Z.
3) إيجاد قيمة المتباينة:

$$\left|R_i - R_j\right| \geq Z\sqrt{\frac{bk(k+1)}{6}}$$

ومن خلال المقارنة لطرفي المتباينة نستنتج أي المعالجات ذات تأثير أكبر من غيرها، وهي سبب الاختلاف.

الجانب العملي

1. الحقيبة البرمجية[4] (Expert System):

إن الهدف من البحث كما تقدم، هو محاولة التغلب على القصور الموجود في الأنظمة الإحصائية الجاهزة، والقصور يتمثل بالآتي: (1) عدم استكمال الاختبارات المطلوبة بعد إجراء الاختبار الرئيسي لمعرفة أيٍّ من المتغيرات البحثية ذات دلالة معنوية. (2) ينبغي صياغة البيانات بشكل يتلاءم مع الاختبار المستخدم،

[1] (القرشي، ص210).
[2] (Glantz, p. 360) و (Siegel, pp. 174 - 178) .
[3] (القرشي، ص15)، (Siegel, p. 180) و (Glantz, p. 361).
[4] ملحق 1

فبالنسبة لاختبار كروسكال- ويلز ينبغي تجميع بيانات العينات المفحوصة في عمود وتحديد عمود آخر يمثل الرمز الخاص بكل قيمة، في حين أن اختبار فريدمان ينبغي إجراء نفس الخطوات السابقة في الاختبار السابق بالإضافة إلى تحديد عمود القطاع (Block) والذي يتم إدخاله من قبل المستخدم. (3) لانجاز المقارنات المتعددة ما بعد الاختبارين المذكورين ينبغي البحث عن قيمة (Z) الجدولية والتعويض عنها في المعادلات الخاصة بالمقارنات المتعددة واستكمال المقارنة يدوياً. ولإنجاز ذلك تم برمجة الخطوات اللازمة لأجراء كل من: (*) اختبار كروسكال - ويلز والمقارنات المتعددة الخاصة به والمطلوب إجراؤها بعد الحصول على قيمة ذلك المختبر ذات القيمة المعنوية، من دون الحاجة لتهيئة البيانات أو البحث عن قيمة (Z) الجدولية. وكذلك: (**) اختبار فريدمان والمقارنات المتعددة له أيضاً. وبالاعتماد على خصائص الحزمة البرمجية (Global Macro) التابعة للنظام الجاهز MINITAB.

1.1 خوارزمية الحقيبة البرمجية:

تم تصميم البرنامج الخبير (الحقيبة البرمجية) وفق خوارزمية، خطواتها الأساسية كالآتي:

1. يتم إدخال البيانات المطلوب إجراء اختبار لها باستخدام اختبار كروسكال - ويلز أو اختبار فريدمان، في الأعمدة (من 1 إلى 5) في النظام الجاهز MINITAB-11 في منصة البيانات (Data Window)، وحسب عدد العينات المبحوثة. إذ أن البرنامج المعد يمكنه أن يجري اختبار لـ: مجموعتين، ثلاث مجاميع، أربع مجاميع أو خمسة مجاميع.
2. يتم تعريف كل من: (1) عدد المجاميع بالتجربة في الخلية الأولى من العمود السادس ((C6(1))، (2) حجم العينة الكلي في الخلية الثانية من العمود السادس ((C6(2))، (3) مستوى الدلالة المعتمد في التجربة (P-Value) في الخلية الثالثة من العمود السادس ((C6(3)).
3. من ثم تنفيذ البرنامج الخبير (Expert System) المعد والذي يحمل الاسم: (CANPT.MAC) وهو اختصار (Compare After Non-Parametric Test) من خلال: منصة الأوامر (Session Window) في النظام الجاهز (Minitab-11) وبعد تفعيل مؤشر النظام، أي: (MTB \ >) في منصة الأوامر، يتم كتابة أيعاز التنفيذ (%)، يتبعه أسم الموقع (Drive)، من ثم اسم البرنامج (File Name)، مع تحديد النوع (Extension) والذي هو (MAC) وهو مختصر لـ (Macros)، وكالآتي: CANPT.MAC % ، إذ يبدأ البرنامج أولاً بالبحث عن قيمة Z الجدولية التي توافق معطيات التجربة. فقد تم تغذية البرنامج بالقيم الجدولية المطلوبة ولأعدادٍ غير محدودة من القيم.
4. عند بدأ التنفيذ للحقيبة البرمجية المعدة سيتم في البدء احتساب قيمة المؤشر الإحصائي كروسكال- ويلز أو فريدمان وحسب ملائمة البيانات لنوع الاختبار والموضحة في الخطوات الواردة فيما تقدم من الجانب النظري.
5. في حالة كون نتيجة الاختبار معنوية، فسيتم البحث من خلال البرنامج المعد عن قيمة (Z) الجدولية والمناسبة لمعطيات التجربة واستخدامها في الخطوات اللاحقة في إجراء المقارنات المتعددة المطلوبة.
6. يقوم البرنامج المعد بطبع النتائج لقيم المؤشر الإحصائي مع إعطاء النتائج لجميع الفروقات المعنوية (للمقارنات المتعددة) بين مجاميع التجربة.

2.1 تطبيق الحقيبة البرمجية:

من أجل الوثوق بإمكانية البرنامج في حل المسائل ذات العلاقة بهدف البحث، تم اختيار المثال التطبيقي (13-2) في صفحة (200)[1] من المصدر (القرشي) لتطبيق اختبار كروسكال- ويلز، وكذلك المثال (14-1) في صفحة (213)[2] من المصدر المذكور لتطبيق اختبار فريدمان على البرنامج الجاهز MINITAB-11تارةً، والحقيبة البرمجية موضوع البحث تارةً أخرى لغرض مقارنة النتائج.

3.1 نتيجة تطبيق اختبار كروسكال – ويلز:

من المثال التطبيقي (13-2) أعلاه كانت النتائج كالآتي:

من خلال البرنامج الجاهز MINITAB-11 :

Kruskal-Wallis Test

Kruskal-Wallis Test on C4

[1] ملحق 2.
[2] ملحق 2.

C5	N	Median	Ave Rank	Z
1	10	305.5	6.9	-3.03
2	6	460.0	15.0	1.55
3	6	729.5	15.7	1.84
Overall	22		11.5	

H = 9.23 DF = 2 P = 0.010

ب. من خلال الحقيبة البرمجية المصممة:

Kruskal-Wallis Test

Kruskal-Wallis Test on C100

C101	N	Median	Ave Rank	Z
1	10	305.5	6.9	-3.03
2	6	460.0	15.0	1.55
3	6	729.5	15.7	1.84
Overall	22		11.5	

H = 9.23 DF = 2 P = 0.010

There is a Sig. between group 1 & 2

There is a Sig. between group 1 & 3

4.1 نتيجة تطبيق اختبار فريدمان:

من المثال التطبيقي (14-1) أعلاه كانت النتائج كالآتي:

أ. من خلال البرنامج الجاهز MINITAB-11 :

Friedman Test

Friedman test for C4 by C7 blocked by C5
S = 22.73 DF = 8 P = 0.004
S = 22.80 DF = 8 P = 0.004 (adjusted for ties)

C7	N	Est Median	Sum of Ranks
1	3	4041	27.0
2	3	1612	23.5
3	3	1488	19.5
4	3	1217	19.0
5	3	829	16.0
6	3	311	12.0
7	3	225	8.0
8	3	200	5.0
9	3	187	5.0

Grand median = 1123

ب. من خلال الحقيبة البرمجية المصممة:

Friedman Test

Friedman test for C100 by C7 blocked by C101

S = 22.73 DF = 8 P = 0.004
S = 22.80 DF = 8 P = 0.004 (adjusted for ties)

C7	N	Est Median	Sum of Ranks
1	3	4041	27.0
2	3	1612	23.5
3	3	1488	19.5
4	3	1217	19.0
5	3	829	16.0
6	3	311	12.0
7	3	225	8.0
8	3	200	5.0
9	3	187	5.0

Grand median = 1123

There is a Sig. between Block 1 & 2.

There is a Sig. between Block 2 & 3.

الاستنتاجات والتوصيات:

1. يمكن استنتاج أن الحقيبة البرمجية المعدة هي أكفأ وأسهل في التطبيق، إذ أن المستخدم في بعض الأحيان يجهل كيفية صياغة البيانات بحيث تلاءم الاختبار المطلوب.
2. يلاحظ أيضا من البرنامج المعد أن المستخدم لا يحتاج للبحث عن قيمة (Z) الجدولية، كما لا يحتاج إلى استكمال المقارنات المتعددة يدوياً، بل يكتفي بإدخال البيانات في أعمدة منفصلة عن بعض وتطبيق البرنامج الخبير المعد ويحصل على كافة النتائج التي يبغيها من البحث.
3. يوصي الباحثان بضرورة تعميم هذا البرنامج وتوسيعه كي يلاءم باقي الاختبارات اللامعلمية الأخرى التي لم تدخل ضمن هذه الحقيبة البرمجية.

المصادر العربية:

1. أبو علام، "التحليل الإحصائي للبيانات باستخدام برنامج spss "، دار النشر للجامعات، ط1، مصر 2003.
2. آدم، أمين إبراهيم، "المبادئ الأساسية الإحصائية في الطرق التطبيقية اللامعلمية"، مطابع الحميضي، مكة المكرمة، ط1، 2005.
3. القرشي، إحسان كاظم: "الطرائق المعلمية و الطرائق اللامعلمية في الاختبارات الإحصائية"، ط 1، مطبعة الديواني ، بغداد، 2007.

المصادر الأجنبية:

1. Christy C.T.: "An Introduction to Non-parametric Statistic for Health Scientists", University of Alberta Health Sciences, Journal. 3(1): PP. 20-26, June, 2006.
2. Glands, S.A.: "Primer of Biostatistics", 4th ed., McGraw-Hill Book Company, N.Y., USA, 1997.
3. Pertie, A.: "Lecture Notes on Medical Statistics", 2nd ed., Blackwell Scientific Publications, USA, 1987.

4. Siegel, S. & Castellan, N.: "Nonparametric Statistics for the Behavioral Sciences", 2nd ed., McGraw-Hill Book Company, N.Y., USA, 1989.
5. Steel, R.G.D. & Torrie, J.H.: "Principles and Procedures of Statistics. A Biometrical Approach", 2nd ed., McGraw-Hill Book Company, N.Y., USA, 1981.

ملحق (1)
الحقيبة البرمجية

```
GMacro
CANPT.MAC.mac
Noecho
Note:    ==========================================
Note:    ==========================================
Note:    ||                                        ||
Note:    ||  This Program is Use To Calculate The "Kruskal-Wallis &  ||
Note:    ||  Friedman Analysis of Variance By Ranks or By Real Data"||
Note:    ||     For The Nonparametric Data in Multiple Groups.       ||
Note:    ||                                        ||
Note:    ==========================================
Note:    ==========================================
# Note: Enter The Data of Samples in C1 To C5
# Note: Enter the number of groups in k1 i.e. "c6(1)"
# Note: Enter the samples size in k2      i.e. "c6(2)"
# Note: Enter the value of Alpha in k3    i.e. "c6(3)"
# Note: Enter (1) one for the Kruskal-Wallis test k50 i.e. "c6(4)=1"
# Note: Enter (2) two for the Friedman       test k50 i.e. "c6(4)=2"
# Note: "let k1= c6(1)   k2= c6(2)  k3= c6(3)   k50= c6(4)
# Note: The value of Z is in k7
# Note: "" The value of Kruskal-Wallis between sample 1 & 2 is in k30""
# Note: "" The value of Kruskal-Wallis between sample 1 & 3 is in k31""
# Note: "" The value of Kruskal-Wallis between sample 2 & 3 is in k32""
# Note: "" The value of Kruskal-Wallis between sample 1 & 4 is in k33""
# Note: "" The value of Kruskal-Wallis between sample 2 & 4 is in k34""
# Note: "" The value of Kruskal-Wallis between sample 3 & 4 is in k35""
# Note: "" The value of Kruskal-Wallis between sample 1 & 5 is in k36""
# Note: "" The value of Kruskal-Wallis between sample 2 & 5 is in k37""
# Note: "" The value of Kruskal-Wallis between sample 3 & 5 is in k38""
# Note: "" The value of Kruskal-Wallis between sample 4 & 5 is in k39""
# Note: Enter the Response Variable of samples in c7
# Note: Enter the Factor Variable of samples in c8
# Note: The Columns   from c9 to c13 is the Ranks of Samples.
# Note: The Constants from k9 to k13 is the Mean of Ranks.
# Note: Enter the Z Value Table in Columns c21-c30
name c20='Z'              let c20(1)=0.0              let c20(2)=0.1
let c20(3)=0.2            let c20(4)=0.3              let c20(5)=0.4
let c20(6)=0.5            let c20(7)=0.6              let c20(8)=0.7
let c20(9)=0.8            let c20(10)=0.9             let c20(11)=1.0
```

let c20(12)=1.1 let c20(13)=1.2 let c20(14)=1.3
let c20(15)=1.4 let c20(16)=1.5 let c20(17)=1.6
let c20(18)=1.7 let c20(19)=1.8 let c20(20)=1.9
let c20(21)=2.0 let c20(22)=2.1 let c20(23)=2.2
let c20(24)=2.3 let c20(25)=2.4 let c20(26)=2.5
let c20(27)=2.6 let c20(28)=2.7 let c20(29)=2.8
let c20(30)=2.9 let c20(31)=3.0 name c21='0.00'
let c21(1)=0.0000 let c21(2)=0.0398 let c21(3)=0.0793
let c21(4)=0.1179 let c21(5)=0.1554 let c21(6)=0.1915
let c21(7)=0.2257 let c21(8)=0.2580 let c21(9)=0.2881
let c21(10)=0.3159 let c21(11)=0.3413 let c21(12)=0.3643
let c21(13)=0.3849 let c21(14)=0.4032 let c21(15)=0.4192
let c21(16)=0.4332 let c21(17)=0.4452 let c21(18)=0.4554
let c21(19)=0.4641 let c21(20)=0.4713 let c21(21)=0.4772
let c21(22)=0.4821 let c21(23)=0.4861 let c21(24)=0.4893
let c21(25)=0.4918 let c21(26)=0.4938 let c21(27)=0.4953
let c21(28)=0.4965 let c21(29)=0.4974 let c21(30)=0.4981
let c21(31)=0.4987 name c22='0.01' let c22(1)=0.0040
let c22(2)=0.0438 let c22(3)=0.0832 let c22(4)=0.1217
let c22(5)=0.1591 let c22(6)=0.1950 let c22(7)=0.2291
let c22(8)=0.2611 let c22(9)=0.2910 let c22(10)=0.3186
let c22(11)=0.3438 let c22(12)=0.3665 let c22(13)=0.3869
let c22(14)=0.4049 let c22(15)=0.4207 let c22(16)=0.4345
let c22(17)=0.4463 let c22(18)=0.4564 let c22(19)=0.4649
let c22(20)=0.4719 let c22(21)=0.4778 let c22(22)=0.4826
let c22(23)=0.4864 let c22(24)=0.4896 let c22(25)=0.4920
let c22(26)=0.4940 let c22(27)=0.4955 let c22(28)=0.4966
let c22(29)=0.4975 let c22(30)=0.4982 let c22(31)=0.4987
name c23='0.02' let c23(1)=0.0080 let c23(2)=0.0478
let c23(3)=0.0871 let c23(4)=0.1255 let c23(5)=0.1628
let c23(6)=0.1985 let c23(7)=0.2324 let c23(8)=0.2642
let c23(9)=0.2939 let c23(10)=0.3212 let c23(11)=0.3461
let c23(12)=0.3686 let c23(13)=0.3888 let c23(14)=0.4066
let c23(15)=0.4222 let c23(16)=0.4357 let c23(17)=0.4474
let c23(18)=0.4573 let c23(19)=0.4656 let c23(20)=0.4726
let c23(21)=0.4783 let c23(22)=0.4830 let c23(23)=0.4868
let c23(24)=0.4898 let c23(25)=0.4922 let c23(26)=0.4941
let c23(27)=0.4956 let c23(28)=0.4967 let c23(29)=0.4976
let c23(30)=0.4982 let c23(31)=0.4987 name c24='0.03'
let c24(1)=0.0120 let c24(2)=0.0517 let c24(3)=0.0910
let c24(4)=0.1293 let c24(5)=0.1664 let c24(6)=0.2019
let c24(7)=0.2357 let c24(8)=0.2673 let c24(9)=0.2967
let c24(10)=0.3238 let c24(11)=0.3485 let c24(12)=0.3708
let c24(13)=0.3907 let c24(14)=0.4082 let c24(15)=0.4236
let c24(16)=0.4370 let c24(17)=0.4484 let c24(18)=0.4582
let c24(19)=0.4664 let c24(20)=0.4732 let c24(21)=0.4788
let c24(22)=0.4834 let c24(23)=0.4871 let c24(24)=0.4901
let c24(25)=0.4925 let c24(26)=0.4943 let c24(27)=0.4957
let c24(28)=0.4968 let c24(29)=0.4977 let c24(30)=0.4983
let c24(31)=0.4988 name c25='0.04' let c25(1)=0.0160

let c25(2)=0.0557 let c25(3)=0.0948 let c25(4)=0.1331
let c25(5)=0.1700 let c25(6)=0.2054 let c25(7)=0.2389
let c25(8)=0.2704 let c25(9)=0.2995 let c25(10)=0.3264
let c25(11)=0.3508 let c25(12)=0.3729 let c25(13)=0.3925
let c25(14)=0.4099 let c25(15)=0.4251 let c25(16)=0.4382
let c25(17)=0.4495 let c25(18)=0.4591 let c25(19)=0.4621
let c25(20)=0.4738 let c25(21)=0.4793 let c25(22)=0.4838
let c25(23)=0.4875 let c25(24)=0.4904 let c25(25)=0.4927
let c25(26)=0.4945 let c25(27)=0.4959 let c25(28)=0.4969
let c25(29)=0.4977 let c25(30)=0.4984 let c25(31)=0.4988
name c26='0.05' let c26(1)=0.0199 let c26(2)=0.0596
let c26(3)=0.0987 let c26(4)=0.1368 let c26(5)=0.1736
let c26(6)=0.2088 let c26(7)=0.2422 let c26(8)=0.2734
let c26(9)=0.3023 let c26(10)=0.3289 let c26(11)=0.3531
let c26(12)=0.3749 let c26(13)=0.3944 let c26(14)=0.4115
let c26(15)=0.4265 let c26(16)=0.4394 let c26(17)=0.4505
let c26(18)=0.4599 let c26(19)=0.4678 let c26(20)=0.4744
let c26(21)=0.4798 let c26(22)=0.4842 let c26(23)=0.4878
let c26(24)=0.4906 let c26(25)=0.4929 let c26(26)=0.4946
let c26(27)=0.4960 let c26(28)=0.4970 let c26(29)=0.4978
let c26(30)=0.4984 let c26(31)=0.4989 name c27='0.06'
let c27(1)=0.0239 let c27(2)=0.0636 let c27(3)=0.1026
let c27(4)=0.1406 let c27(5)=0.1770 let c27(6)=0.2123
let c27(7)=0.2454 let c27(8)=0.2764 let c27(9)=0.3051
let c27(10)=0.3315 let c27(11)=0.3554 let c27(12)=0.3770
let c27(13)=0.3962 let c27(14)=0.4131 let c27(15)=0.4279
let c27(16)=0.4406 let c27(17)=0.4515 let c27(18)=0.4608
let c27(19)=0.4686 let c27(20)=0.4750 let c27(21)=0.4803
let c27(22)=0.4846 let c27(23)=0.4881 let c27(24)=0.4909
let c27(25)=0.4931 let c27(26)=0.4948 let c27(27)=0.4961
let c27(28)=0.4971 let c27(29)=0.4979 let c27(30)=0.4985
let c27(31)=0.4989 name c28='0.07' let c28(1)=0.0279
let c28(2)=0.0675 let c28(3)=0.1064 let c28(4)=0.1443
let c28(5)=0.1808 let c28(6)=0.2157 let c28(7)=0.2486
let c28(8)=0.2794 let c28(9)=0.3078 let c28(10)=0.3340
let c28(11)=0.3577 let c28(12)=0.3790 let c28(13)=0.3980
let c28(14)=0.4147 let c28(15)=0.4292 let c28(16)=0.4418
let c28(17)=0.4525 let c28(18)=0.4616 let c28(19)=0.4693
let c28(20)=0.4756 let c28(21)=0.4808 let c28(22)=0.4850
let c28(23)=0.4884 let c28(24)=0.4911 let c28(25)=0.4932
let c28(26)=0.4949 let c28(27)=0.4962 let c28(28)=0.4972
let c28(29)=0.4979 let c28(30)=0.4985 let c28(31)=0.4989
name c29='0.08' let c29(1)=0.0319 let c29(2)=0.0174
let c29(3)=0.1103 let c29(4)=0.1480 let c29(5)=0.1844
let c29(6)=0.2190 let c29(7)=0.2517 let c29(8)=0.2823
let c29(9)=0.3106 let c29(10)=0.3365 let c29(11)=0.3599
let c29(12)=0.3810 let c29(13)=0.3997 let c29(14)=0.4162
let c29(15)=0.4306 let c29(16)=0.4429 let c29(17)=0.4535
let c29(18)=0.4625 let c29(19)=0.4699 let c29(20)=0.4761
let c29(21)=0.4812 let c29(22)=0.4854 let c29(23)=0.4887

```
let c29(24)=0.4913
let c29(25)=0.4934
let c29(26)=0.4951
let c29(27)=0.4963
let c29(28)=0.4973
let c29(29)=0.4980
let c29(30)=0.4986
let c29(31)=0.4990
name c30='0.09'
let c30(1)=0.0359
let c30(2)=0.0753
let c30(3)=0.1141
let c30(4)=0.1517
let c30(5)=0.1879
let c30(6)=0.2224
let c30(7)=0.2549
let c30(8)=0.2852
let c30(9)=0.3133
let c30(10)=0.3389
let c30(11)=0.3621
let c30(12)=0.3830
let c30(13)=0.4015
let c30(14)=0.4177
let c30(15)=0.4319
let c30(16)=0.4441
let c30(17)=0.4545
let c30(18)=0.4633
let c30(19)=0.4706
let c30(20)=0.4767
let c30(21)=0.4817
let c30(22)=0.4857
let c30(23)=0.4890
let c30(24)=0.4916
let c30(25)=0.4936
let c30(26)=0.4952
let c30(27)=0.4964
let c30(28)=0.4974
let c30(29)=0.4981
let c30(30)=0.4986
let c30(31)=0.4990
let k1=c6(1)
let k2=c6(2)
let k3=c6(3)
let k4=k3/(k1*(k1-1))
let k5=0.5-k4
let k50=c6(4)
do k6=1:31
 if k5=c21(k6)
let k7=c20(k6)+0.00
endif
 enddo
erase k6
 do k6=1:31
 if k5=c22(k6)
let k7=c20(k6)+0.01
endif
 enddo
erase k6
 do k6=1:31
 if k5=c23(k6)
let k7=c20(k6)+0.02
endif
 enddo
erase k6
 do k6=1:31
 if k5=c24(k6)
let k7=c20(k6)+0.03
endif
 enddo
erase k6
 do k6=1:31
 if k5=c25(k6)
let k7=c20(k6)+0.04
endif
enddo
erase k6
 do k6=1:31
 if k5=c26(k6)
let k7=c20(k6)+0.05
endif
enddo
erase k6
 do k6=1:31
 if k5=c27(k6)
let k7=c20(k6)+0.06
endif
enddo
erase k6
 do k6=1:31
 if k5=c28(k6)
let k7=c20(k6)+0.07
endif
enddo
erase k6
 do k6=1:31
 if k5=c29(k6)
let k7=c20(k6)+0.08
endif
enddo
erase k6
 do k6=1:31
 if k5=c30(k6)
let k7=c20(k6)+0.09
else
let k7=0.303395
endif
 enddo
 print k7
```

```
Note:     ================================
Note:     ================================
Note:     ||                                ||
Note:     || Start the Non-Parametric Tests Program    ||
Note:     ||                                ||
Note:     ================================
Note:     ================================
if k1=3
count c1 k21.
count c2 k22.
count c3 k23.
stack c1-c3 c100;
Subscripts c101.
rank c100 c102.
unstack c102 c9 c10 c11;
Subscripts c101.
Mean c9 k9.
Mean c10 k10.
Mean c11 k11.
Let k40 = abso(k9-k10)
Let k41 = abso(k9-k11)
Let k42 = abso(k10-k11)
Let k30=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k22)))
let k31=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k23)))
let k32=k7* sqrt((k2*(k2+1)/12)*((1/k22)+(1/k23)))
if k50 = 1              goto 1              elseif k50 = 2
goto 2                  else                goto 3
endif                   mlabel 1            Note:
Kruskal-Wallis c100 c101.                   Note:
if k40>k30          Note: " There is a Sig. Between Group 1 & 2 "
Elseif k41>k31      Note: " There is a Sig. Between Group 1 & 3 "
Elseif k42>k32      Note: " There is a Sig. Between Group 2 & 3 "
Else
Note:     ================================
Note:     ================================
Note:     ||                                ||
Note:     || There are No Sig. Between All Groups  ||
Note:     ||                                ||
Note:     ================================
Note:     ================================
Endif                   goto 5              mlabel 2
rank c1 c201.           rank c2 c202.       rank c3 c203.
set c7                  k1(1:k21)           end
Sum c201 k51.           Sum c202 k52.       Sum c203 k53.
let k55 = abso(k51-k52)                     let k56 = abso(k51-k53)
let k57 = abso(k52-k53)                     let k54=k7*
sqrt(k21*k1*(k1-1))
Note:
Friedman c100 c7 c101
Note:
```

```
if k55>k54
Note: " There is a Sig. between Block 1 & 2 "
Elseif k56>k54                    Note: " There is a Sig. between
Block 1 & 3 "
Elseif k57>k54                    Note: " There is a Sig. between
Block 2 & 3 "
else
Note:    ============================
Note:    ============================
Note:    ||                          ||
Note:    || There are No Sig. Between All Blocks  ||
Note:    ||                          ||
Note:    ============================
Note:    ============================
Endif                    goto 5                    endif
if k1=4                  count c1 k21.             count c2 k22.
count c3 k23.            count c4 k24.             stack c1-c4 c100;
subscripts c101.         rank c100 c102.           unstack c102 c9
c10 c11 c12;
subscripts c101.         mean c9 k9.               mean c10 k10.
mean c11 k11.            mean c12 k12.             let k40 = abso(k9-
k10)
let k41 = abso(k9-k11)   let k42 = abso(k9-k12)    let k43 = abso(k10-
k11)
let k44 = abso(k10-k12)  let k45 = abso(k11-k12)
let k30=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k22)))
let k31=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k23)))
let k32=k7* sqrt((k2*(k2+1)/12)*((1/k22)+(1/k23)))
let k33=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k24)))
let k34=k7* sqrt((k2*(k2+1)/12)*((1/k22)+(1/k24)))
let k35=k7* sqrt((k2*(k2+1)/12)*((1/k23)+(1/k24)))
if k50 = 1               goto 6                    elseif k50 = 2
goto 7                   else                      goto 8
endif                    mlabel 6                  Note:
Kruskal-Wallis c100 c101.  Note:                   if k40>k30
Note: " There is a Sig. between group 1 & 2 "      Elseif k41>k31
Note: " There is a Sig. between group 1 & 3 "      Elseif k42>k33
Note: " There is a Sig. between group 1 & 4 "      Elseif k43>k32
Note: " There is a Sig. between group 2 & 3 "      Elseif k44>k34
Note: " There is a Sig. between group 2 & 4 "      Elseif k45>k35
Note: " There is a Sig. between group 3 & 4 "      else
Note:    ============================
Note:    ============================
Note:    ||                          ||
Note:    || There are No Sig. Between All Groups ||
Note:    ||                          ||
Note:    ============================
Note:    ============================
Endif                    goto 15                   mlabel 7
rank c1 c201.            rank c2 c202.             rank c3 c203.
```

```
rank c4 c204.                set c7                       k1(1:k21)
end                          Sum c201 k51.                Sum c202 k52.
Sum c203 k53.                Sum c204 k54.                let k55 =
abso(k51-k52)
let k56 = abso(k51-k53)      let k57 = abso(k51-k54)      let k58 =
abso(k52-k53)
let k59 = abso(k52-k54)      let k60 = abso(k53-k54)
let k70=k7* sqrt(k21*k1*(k1-1))                           Note:
Friedman c100 c7 c101        Note:                        if k55>k70
Note: " There is a Sig. between Block 1 & 2 "             Elseif k56>k70
Note: " There is a Sig. between Block 1 & 3 "             Elseif k57>k70
Note: " There is a Sig. between Block 1 & 4 "             Elseif k58>k70
Note: " There is a Sig. between Block 2 & 3 "             Elseif k59>k70
Note: " There is a Sig. between Block 2 & 4 "             Elseif k60>k70
Note: " There is a Sig. between Block 3 & 4 "             else
Note:    ============================
Note:    ============================
Note:    ||                          ||
Note:    || There are No Sig. Between All Blocks  ||
Note:    ||                          ||
Note:    ============================
Note:    ============================
Endif                        goto 15                      endif
if k1= 5                     count c1 k21.                count c2 k22.
count c3 k23.                count c4 k24.                count c5 k25.
stack c1-c5 c100;            subscripts c101.             rank c100 c102.
unstack c102 c9 c10 c11 c12 c13;                          subscripts c101.
mean c9 k9.                  mean c10 k10.                mean c11 k11.
mean c12 k12.                mean c13 k13.                let k40 = abso(k9-
k10)
let k41 = abso(k9-k11)       let k42 = abso(k9-k12)       let k43 = abso(k9-
k13)
let k44 = abso(k10-k11)      let k45 = abso(k10-k12)      let k46 = abso(k10-
k13)
let k47 = abso(k11-k12)      let k48 = abso(k11-k13)      let k49 = abso(k12-
k13)
let k30=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k22)))
let k31=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k23)))
let k32=k7* sqrt((k2*(k2+1)/12)*((1/k22)+(1/k23)))
let k33=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k24)))
let k34=k7* sqrt((k2*(k2+1)/12)*((1/k22)+(1/k24)))
let k35=k7* sqrt((k2*(k2+1)/12)*((1/k23)+(1/k24)))
let k36=k7* sqrt((k2*(k2+1)/12)*((1/k21)+(1/k25)))
let k37=k7* sqrt((k2*(k2+1)/12)*((1/k22)+(1/k25)))
let k38=k7* sqrt((k2*(k2+1)/12)*((1/k23)+(1/k25)))
let k39=k7* sqrt((k2*(k2+1)/12)*((1/k24)+(1/k25)))
if k50 = 1                   goto 9                       elseif k50 = 2
goto 10                      else                         goto 11
endif                        mlabel 9                     Note:
Kruskal-Wallis c100 c101.    Note:                        if k40>k30
```

```
Note: " There is a Sig. between group 1 & 2 "          Elseif k41>k31
Note: " There is a Sig. between group 1 & 3 "          Elseif k42>k33
Note: " There is a Sig. between group 1 & 4 "          Elseif k43>k36
Note: " There is a Sig. between group 1 & 5 "          Elseif k44>k32
Note: " There is a Sig. between group 2 & 3 "          Elseif k45>k34
Note: " There is a Sig. between group 2 & 4 "          Elseif k46>k37
Note: " There is a Sig. between group 2 & 5 "          Elseif k47>k35
Note: " There is a Sig. between group 3 & 4 "          Elseif k48>k38
Note: " There is a Sig. between group 3 & 5 "          Elseif k49>k39
Note: " There is a Sig. between group 4 & 5 "          else
Note:    ==============================
Note:    ==============================
Note:    ||                            ||
Note:    || There are No Sig. Between All Groups ||
Note:    ||                            ||
Note:    ==============================
Note:    ==============================
Endif                          goto 25                 mlabel 10
rank c1 c201.                   rank c2 c202.           rank c3 c203.
rank c4 c204.                   rank c5 c205.           set c7
k1(1:k21)                       end                     Sum c201 k51.
Sum c202 k52.                   Sum c203 k53.           Sum c204 k54.
Sum c205 k55.                   let k56 = abso(k51-k52)  let k57 =
abso(k51-k53)
let k58 = abso(k51-k54)         let k59 = abso(k51-k55)  let k60 =
abso(k52-k53)
let k61 = abso(k52-k54)         let k62 = abso(k52-k55)  let k63 =
abso(k53-k54)
let k64 = abso(k53-k55)         let k65 = abso(k54-k55)
let k70=k7* sqrt(k21*k1*(k1-1))                         Note:
Friedman c100 c7 c101           Note:                    if k56>k70
Note: " There is a Sig. between Block 1 & 2 "            Elseif k57>k70
Note: " There is a Sig. between Block 1 & 3 "            Elseif k58>k70
Note: " There is a Sig. between Block 1 & 4 "            Elseif k59>k70
Note: " There is a Sig. between Block 1 & 5 "            Elseif k60>k70
Note: " There is a Sig. between Block 2 & 3 "            Elseif k61>k70
Note: " There is a Sig. between Block 2 & 4 "            Elseif k62>k70
Note: " There is a Sig. between Block 2 & 5 "            Elseif k63>k70
Note: " There is a Sig. between Block 3 & 4 "            Elseif k64>k70
Note: " There is a Sig. between Block 3 & 5 "            Elseif k65>k70
Note: " There is a Sig. between Block 4 & 5 "            Else
Note:    ==============================
Note:    ==============================
Note:    ||                            ||
Note:    || There are No Sig. Between All Blocks  ||
Note:    ||                            ||
Note:    ==============================
Note:    ==============================
Endif                          goto 25                  endif
mlabel 3                        mlabel 8                 mlabel 11
```

```
Note:    ==========================================
Note:    ==========================================
Note:    ||                                      ||
Note:    || The Value of c6(4) is False.         ||
Note:    || The True Value either " One " for the Kruskal-Wallis Test.||
Note:    || or " Two " for the Friedman Test.    ||
Note:    ||                                      ||
Note:    ==========================================
Note:    ==========================================
goto 25                  mlabel 5                  mlabel 15
mlabel 25
endmacro
```

ملحق (2)

المثالان المطبقان في الحقيبة البرمجية

1. مثال (13-2):

البيانات الآتية تمثل ثلاث مجموعات تم اعطائهم نوع من الدواء واجري قياس لتأثير ذلك الدواء في الدم. المطلوب اختبار كروسكال- ويلز لبيان اختلاف المجموعات الثلاث لمستوى معنوية 5%.

المجموعة الأولى	262	307	211	323	454	339	304	154	287	356
المجموعة الثانية	465	501	455	355	468	362				
المجموعة الثالثة	343	772	207	1048	838	687				

2- مثال (14-1):

تم قياس تأثير نوع من الدواء على 9 عينات بطرائق ثلاث. المطلوب اختبار الفرضية القائلة بعدم وجود فرق بين تلك الطرائق لمستوى معنوية 0.005 للبيانات الآتية:

الطرائق\القطاعات	1	2	3	4	5	6	7	8	9
الطريقة A	4000	1600	1600	1200	840	352	224	200	189
الطريقة B	3210	1040	647	570	445	156	155	99	70
الطريقة C	6120	2410	2210	2060	1400	249	224	208	227

.....

.....

.....