



الجامعة الافتراضية السورية  
SYRIAN VIRTUAL UNIVERSITY

# مقرر المترجمات

الفصل الأول: المقدمة

**I**nformation

**T**echnology

**E**ngineering



Powered by:



العنوان	رقم الصفحة
1. مقدمة	3
2. ماذا تعني الترجمة	4
3. أنواع المترجمات	6
4. اختلاف المترجمات	7

## كلمات مفتاحية

المترجم، مترجم أحادي المرور، مترجم متعدد المرور.

## ملخص

سنستعرض في هذا الفصل أساسيات في عملية الترجمة، ومراحل تنفيذ برنامج برمجي.

## أهداف تعليمية

- الترجمة.
- أنواع المترجمات.
- سبب اختلاف المترجمات.

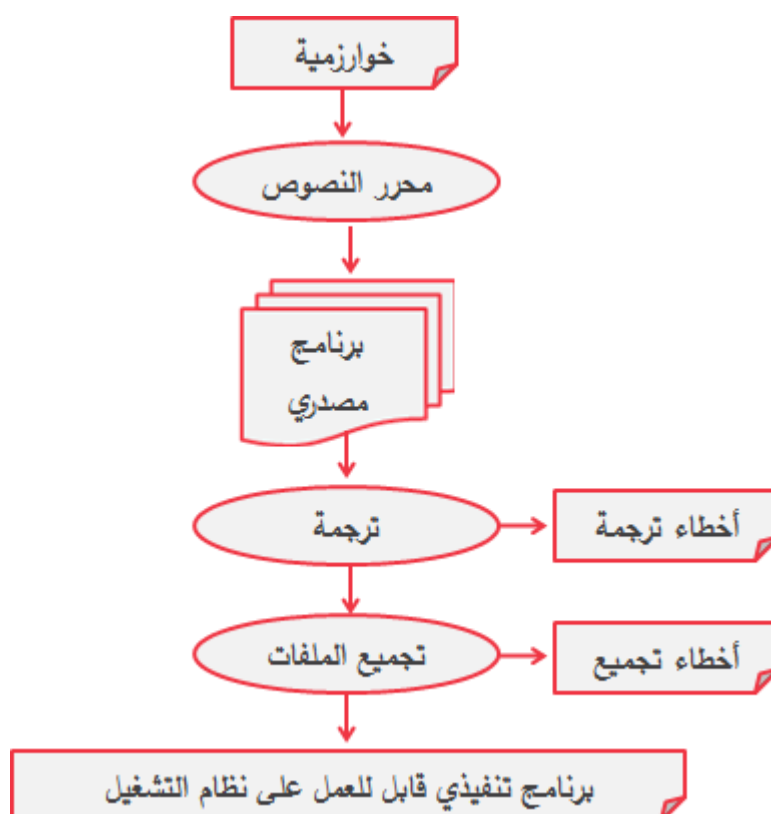
## 1. مقدمة

إن النص البرمجي لإحدى لغات البرمجة مكتوبٌ باستخدام كلمات ورموز بالحروف الإنجليزية. لذلك يكون النص البرمجي مقروء ومفهوم من قبلنا. لكن عرض نفس البرنامج على المعالج لن يؤدي لأي نتيجةٍ مشابهة. لا يمكن للمعالج فهم الحروف والرموز والتعابير المكتوبة باللغة الإنجليزية بشكلٍ مباشر، فهو يفهم لغةً وحيدة: اللغة الثنائية المؤلفة من سلاسل أصفار ووحدان. فيجب العمل على أن تكون البرامج التي نقوم بكتابتها مفهومةً بالنسبة للمعالج، فإننا نحتاج لمن يفسرها له ويشرح ما الذي نريده منه بالضبط. فالبرنامج بالنتيجة ليس سوى مجموعة من التعليمات التي نريد المعالج أن يقوم بتنفيذها. بعالم لغات البرمجة يبرز مصطلح المترجم "Compiler" للإشارة إلى البرنامج الوسيط الذي يأخذ على عاتقه مهمة جعل النصوص البرمجية المكتوبة بإحدى لغات البرمجة عالية المستوى مفهومةً بالنسبة للمعالج.

## 2. ماذا تعني الترجمة

يستخدم أي مبرمج أداة ضرورية جداً في عملية البرمجة، ندعوها المترجم. يمكننا تعريف المترجم بأنه برنامج حاسوبي يترجم النص البرمجي الذي نكتبه بلغة برمجة عالية المستوى (C, Pascal, C++, C#, Java, ...etc) إلى مجموعة تعليمات قابلة للتنفيذ من قبل الحاسوب. تكون هذه التعليمات التنفيذية مكتوبة بلغة منخفضة المستوى سواء كانت لغة ثنائية مؤلفة من أصفار ووحدان (0,1)، أو لغة تجميع.

مثال عن بعض المترجمات: jdk هو مترجم للغة java، visual studio مترجم للغات (c, c++, c#) تمر عملية تشغيل برنامج حاسوبي بمجموعة من المراحل التي نمثلها في الشكل التالي:



تجري كتابة النص البرمجي (أو النصوص البرمجية) لأي برنامج حاسوبي، باستخدام محرر نصوص (ضمن ملف واحد أو ضمن مجموعة من الملفات). ندعو هذه النصوص البرمجية التي تؤلف برنامج حاسوبي، بالبرنامج المصدري (Source Program).

يمر البرنامج المصدري بعد ذلك، بمرحلة ترجمة (Compilation) وتجميع (Linking) يجري فيها ربط الملفات الحاوية على البرنامج المصدري الواحد ببعضها البعض وترجمتها إلى مجموعة من التعليمات التنفيذية المكتوبة بلغة منخفضة المستوى.

تشكل التعليمات التنفيذية الناتجة برنامج جديد ندعوه البرنامج الهدف (Destination Program). يأخذ شكل ملف تنفيذي قابل للتشغيل مباشرةً على نظام التشغيل.

توجد أيضاً مترجمات تترجم من لغة عالية المستوى high- level language إلى لغة أخرى مختلفة عالية المستوى، أو تترجم في بعض الأحيان إلى لغة وسيطة تحتاج إلى معالجة إضافية.

تظهر خلال مراحل الترجمة والتجميع أخطاء ندعوها أخطاء الترجمة (تكون ناجمة عن أخطاء في نصوص البرنامج المصدري) أو أخطاء التجميع (تكون ناجمة عن أخطاء في ربط الملفات الحاوية على النصوص). تؤدي هذه الأخطاء إلى توقف عملية الترجمة حتى يجري تصحيحها من قبل المبرمج، قبل إعادة تشغيل المترجم لتوليد "برنامج هدف" خال من الأخطاء.

### 3. أنواع المترجمات

- مترجم أحادي المرور: one-pass compiler تتم كامل عملية التحليل في هذا النوع من المترجمات بمرور واحد على البرنامج المصدري. لذلك هذا النوع سريع جداً كما في المترجمات البدائية الخاصة بلغة باسكال. ويستخدم هذا النوع من المترجمات وفقاً لتصميم اللغة وطبقاً لقواعد محددة موضوعة للتصريح عن المتغيرات والأغراض الأخرى المستخدمة.
- مترجم متعدد المرور: multi-pass compiler يمكن أن تتم كل مرحلة من مراحل التحليل بأكثر من مرور على البرنامج المصدري.

## 4. اختلاف المترجمات

تجدر الإشارة إلى أن عملية بناء المترجم (والذي عرّفناه كبرنامج حاسوبي) تتعلق بعنصرين اثنين بآن واحد:

1. لغة البرمجة المصدرية عالية المستوى الذي يستخدمها المبرمج.

2. نظام التشغيل الذي سيجري تشغيل البرنامج عليه.

فعلى سبيل المثال، يختلف مترجم لغة C++ الذي يعمل على نظام Windows عن مترجم لغة C++ الذي يعمل على نظام Linux، نظراً لضرورة توليد تعليمات تشغيل تنفيذية مختلفة في هاتين الحالتين، بالرغم من أننا نتكلم عن نفس اللغة وهي C++. في حين، يختلف مترجم لغة C++ عن مترجم لغة Pascal حتى ولو كان المترجمان يعملان على نظام Windows، نظراً لأننا نترجم لغتين برمجتين مختلفتين.