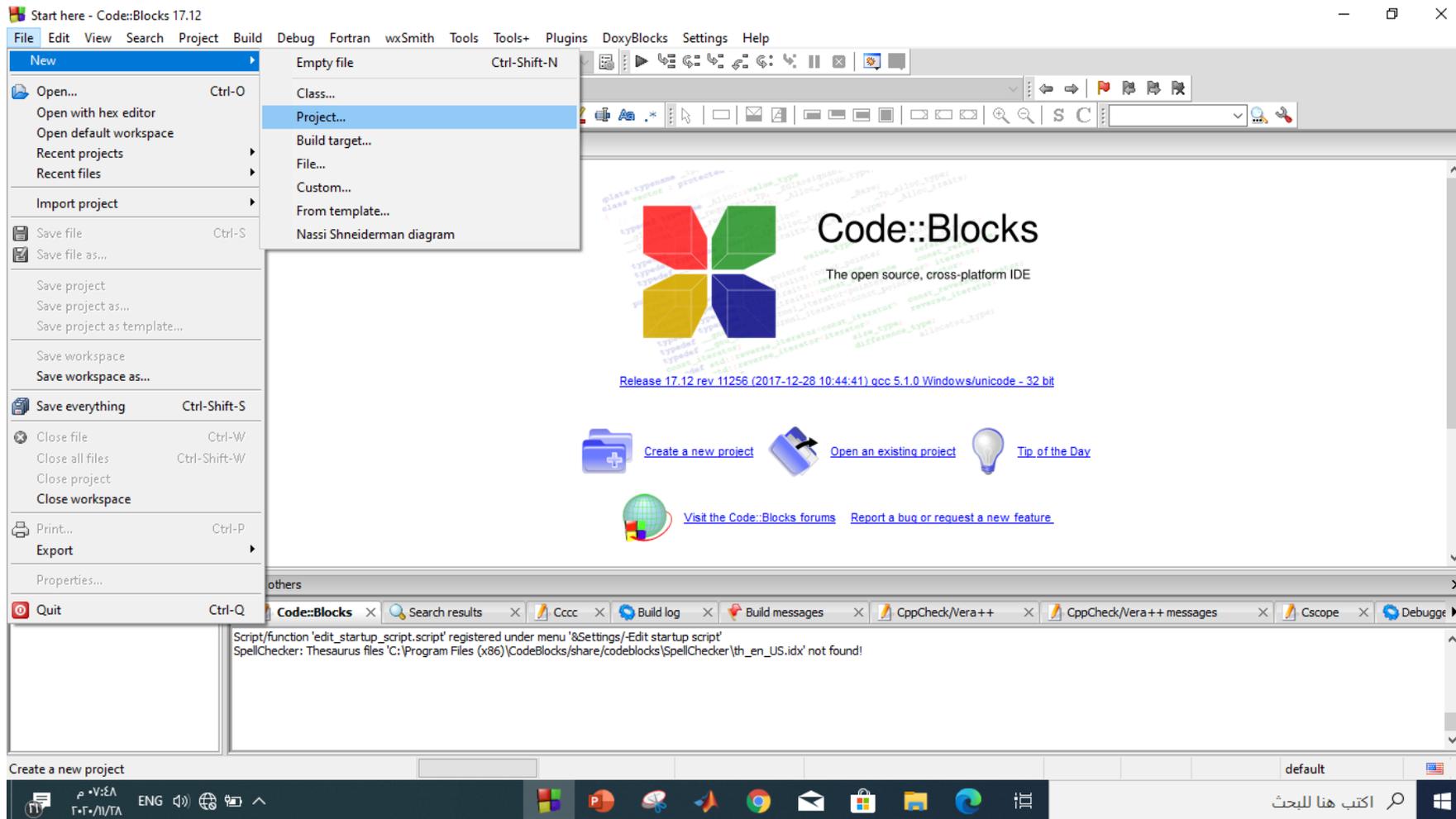




قسم الهندسة المعلوماتية
مقرر مدخل إلى الخوارزميات والبرمجة

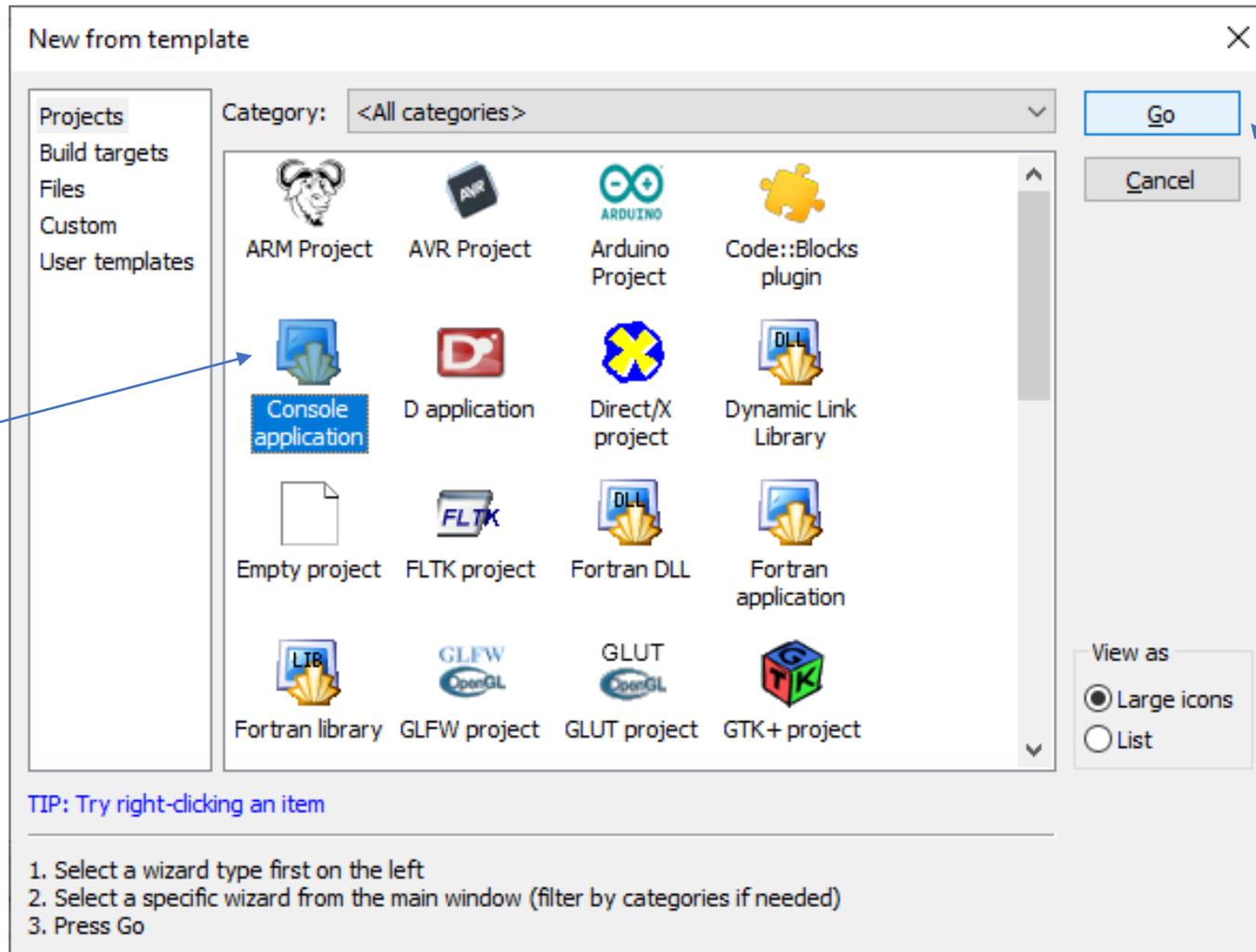
محاضرات الأسبوع 5 الفصل الثاني

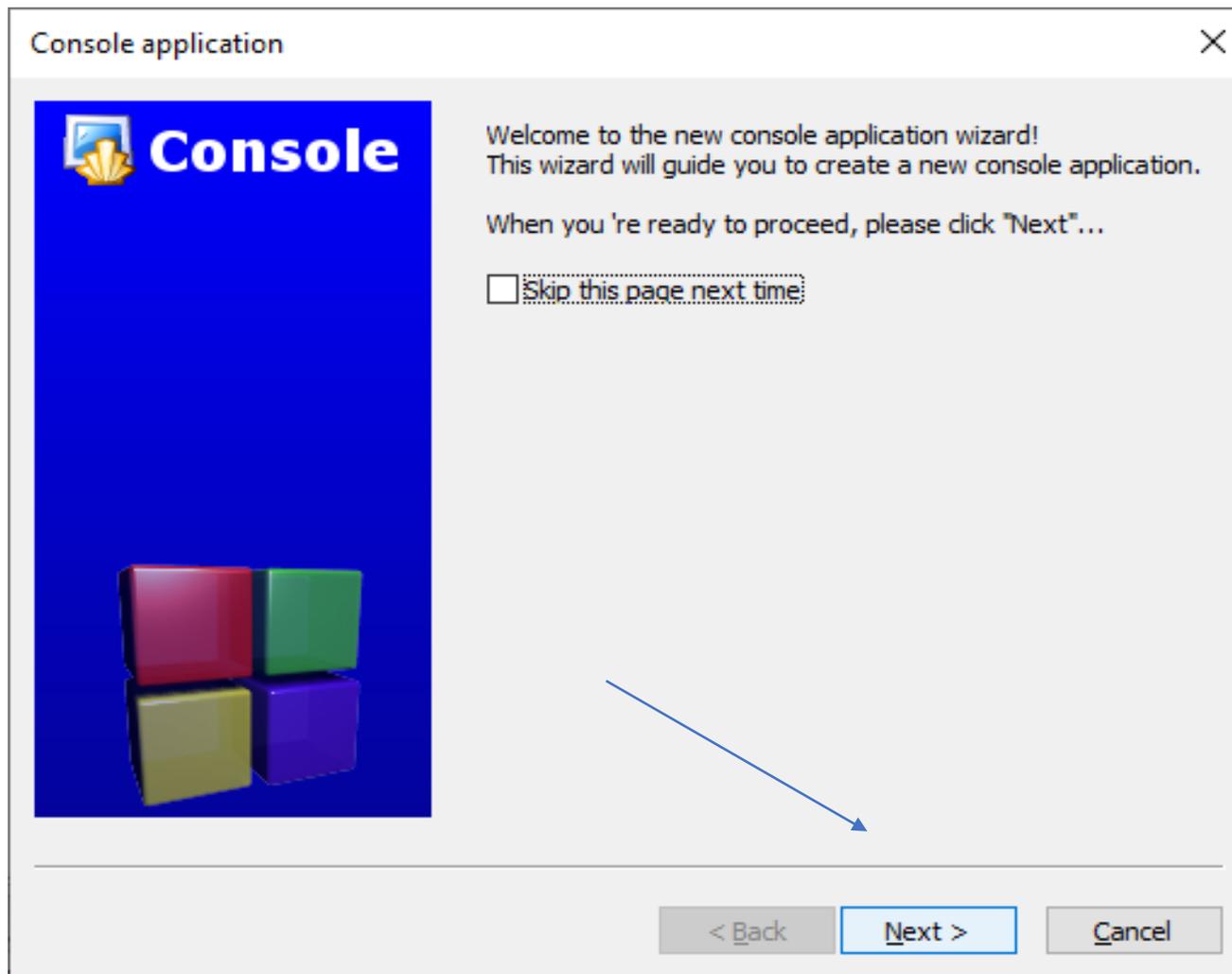
2023/2022



بعد تنصيب البرنامج
بالنقر المزدوج على
الايقونة الخاصة به
تظهر الواجهة التالية

فيما يلي خطوات
إنشاء مشروع جديد
في بيئة code::blocks





Console application

Please select the folder where you want the new project to be created as well as its title.

Project title:
ERT

Folder to create project in:

Project filename:
ERT.cbp

Resulting filename:
<invalid path>

< Back Next > Cancel

الاستعراض بحثاً عن مجلد

Please select the folder to create your project in

الصور <
الفيديو <
المستندات >
المستندات >
MATLAB
قوالب Office المخصصة
CODEBLOCKS

المجلد:
مجلد جديد

إنشاء مجلد جديد موافق إلغاء الأمر

Console application

 **Console**

Please select the compiler to use and which configurations you want enabled in your project.

Compiler:
GNU GCC Compiler

Create "Debug" configuration: Debug

"Debug" options
Output dir.: bin\Debug
Objects output dir.: obj\Debug

Create "Release" configuration: Release

"Release" options
Output dir.: bin\Release
Objects output dir.: obj\Release

< Back Finish Cancel

[TYU] - Code::Blocks 17.12

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

Debug

Management

Projects Symbols Files

Workspace

TYU

Sources

main.cpp

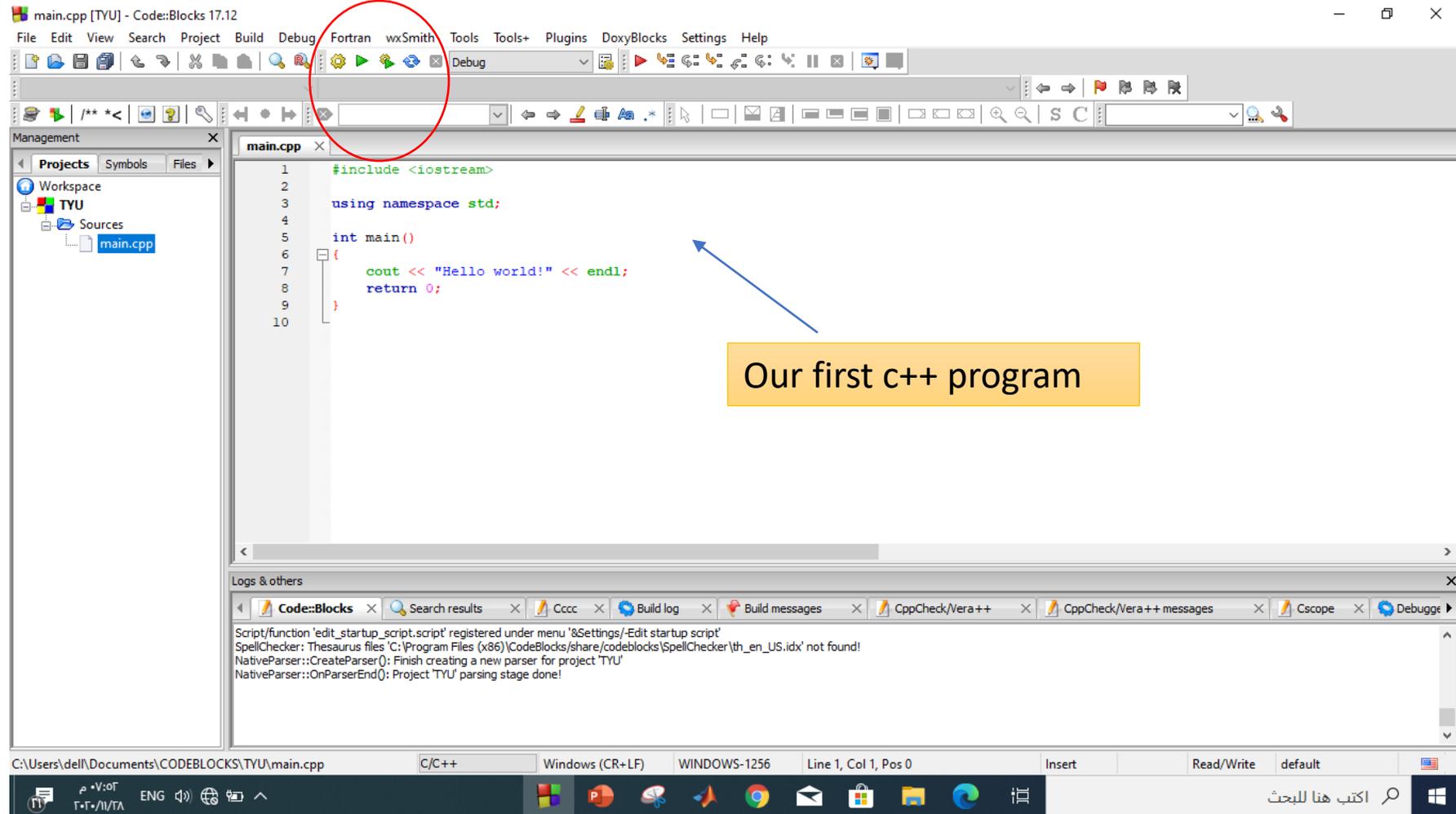
Logs & others

Code::Blocks Search results Cccc Build log Build messages CppCheck/Vera++ CppCheck/Vera++ messages Cscope Debugge

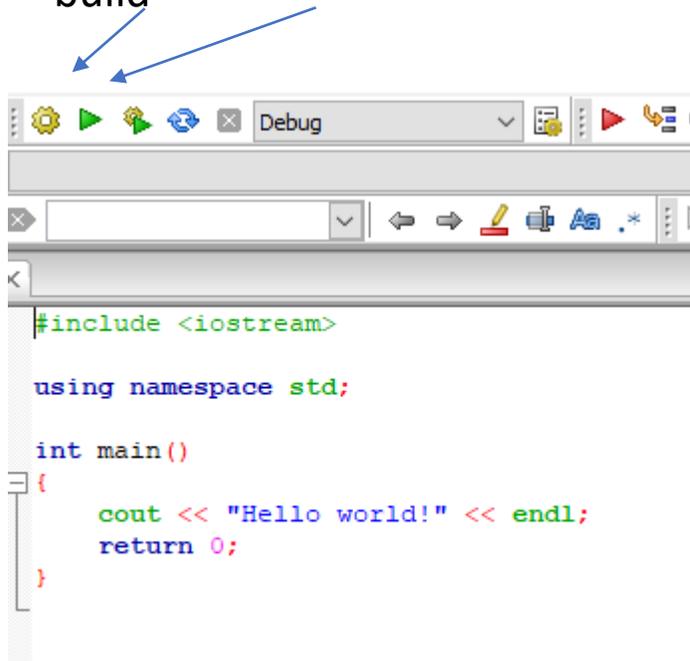
Script/function 'edit_startup_script.script' registered under menu '&Settings/-Edit startup script'
SpellChecker: Thesaurus files 'C:\Program Files (x86)\CodeBlocks\share\codeblocks\SpellChecker\th_en_US.idx' not found!

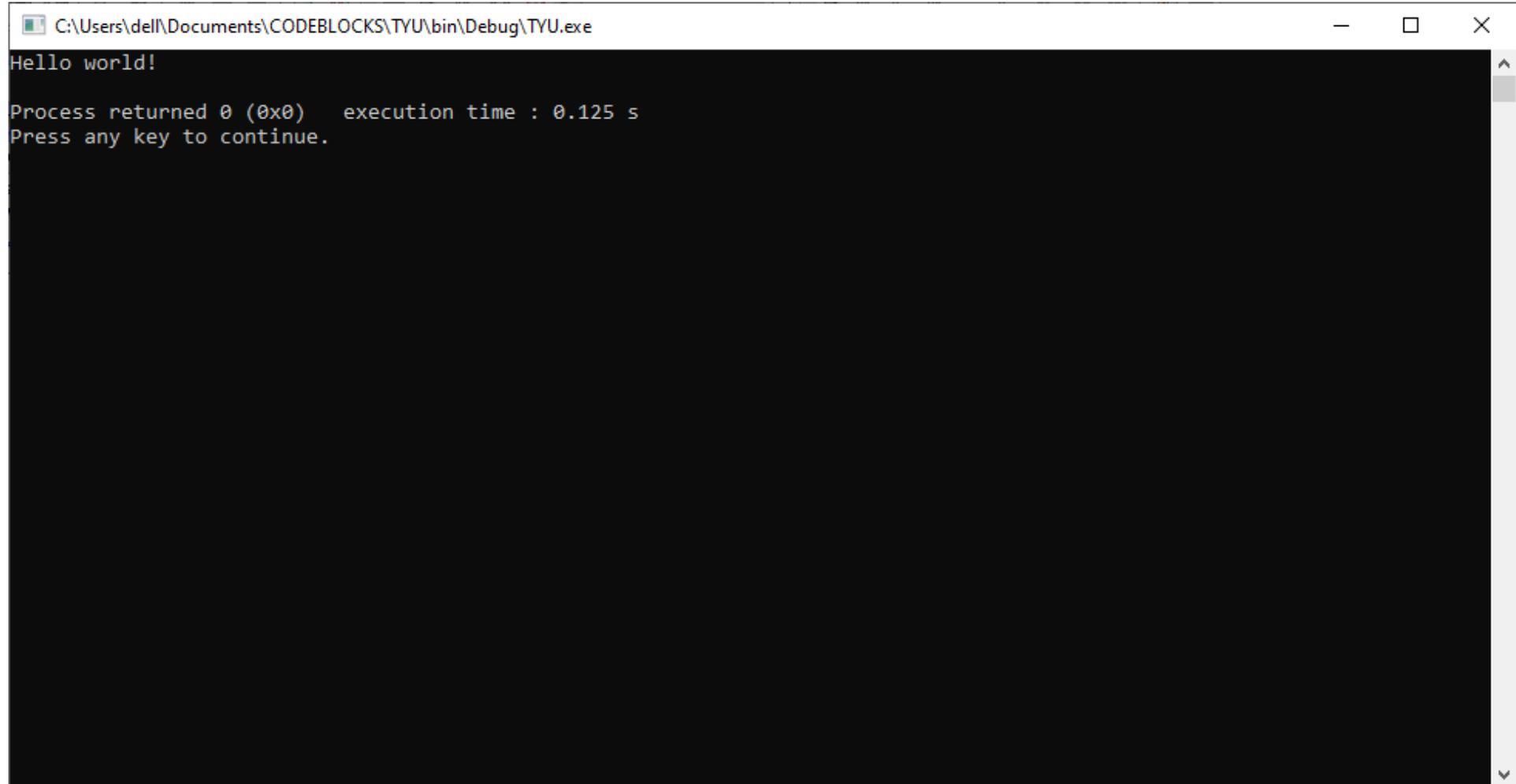
Welcome to Code::Blocks!

default



build run





C:\Users\dell\Documents\CODEBLOCKS\TYU\bin\Debug\TYU.exe

```
Hello world!  
Process returned 0 (0x0)   execution time : 0.125 s  
Press any key to continue.
```

مقدمة في لغة C++

هيكلية البرنامج في لغة C++ :

- التوجيه لمرحلة ما قبل المعالجة ويرمز لهذه العملية بـ # حيث يتم التوجيه لتضمين include المكتبة القياسية للدخل والخرج `iostream`.
- تعريف فضاء الأسماء `namespace` وهو الفضاء `std` ويتضمن جميع أسماء الأدوات في اللغة، أي الكلمات المحجوزة.



قسم التضمينات

```
#include<iostream>
using namespace std;
```

التابع الرئيسي

```
int main()
{
    الأوامر والتعليمات التي تنجز حل المسألة
    return 0;
}
```

مقدمة في لغة C++

قسم التضمينات	<pre>#include<iostream> using namespace std;</pre>
التابع الرئيسي	<pre>int main() { الأوامر والتعليمات التي تنجز حل المسألة return 0; }</pre>

- ملاحظة ١: ان لغة c++ حساسة لحالة الاحرف الصغيرة والكبيرة
- ملاحظة ٢: يمكن وضع أكثر من تعليمة على سطر واحد شرط أن تنتهي كل تعليمة بفاصلة منقوطة

٣. التصريح عن التابع الرئيسي () main وتحديد نوع القيمة المعادة منه وهنا حتما int أي قيمة صحيحة integer. واسم التابع الرئيسي كلمة محجوزة يبدأ منها تنفيذ تعليمات البرنامج.

٤. {} يكتب بين هذان القوسان التعليمات البرمجية التي يتضمنها البرنامج. ويجب انهاء كل تعليمة داخلها بفاصلة منقوطة للدلالة على انتهاء التعليمة.

٥. return 0 وهي كلمة محجوزة تقوم بإيقاف تنفيذ البرنامج وإرجاع القيمة التي تليها وهنا . قيمة صحيحة int للدلالة على انهاء تنفيذ التعليمات بنجاح ضمن التابع الرئيسي main.

الكلمات المحجوزة في لغة ++c للخروج والدخول

للإخراج: `cout << .. << endl;`

- عادة تستخدم للطباعة والإخراج على الشاشة
- يجب فصل كل عمليات الطباعة من أجل كل `cout` واحدة بإشارة `<<`
- تطبع قيمة المتحول أو ناتج العملية الحسابية، وأما ما بين فاصلتين علويتين يطبع كما هو كرسالة دون أي تغيير.
- عادة تنتهي عملية الإخراج بالنزول لسطر جديد لتنسيق عملية الخرج على الشاشة باستخدام `endl`.

للإدخال: `cin >> .. >> ;`

- عادة تستخدم للإدخال من لوحة المفاتيح.
- يجب فصل كل عمليات الإدخال من أجل كل `cin` واحدة بإشارة `>>`

افتح برنامج code::blocks وقم بفتح مشروع جديد ضمنه واحفظه باسم تختاره واكتب البرنامج التالي :

انتبه:

يتضمن هذا البرنامج تعليمة لطباعة رسالة !helloworld لذا تم وضعها ضمن فاصلتين علوية

تم استخدام عبارة الإخراج القياسية <<cout .

الكلمة المحجوزة endl تقوم بالنزول لسطر جديد على الشاشة .

القيمة 0 التي تم اعادتها في تعليمة return ضمن التابع main للدلالة على انتهاء التنفيذ بنجاح، ويمكن تغيير هذه القيمة إلى أي عدد صحيح.

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Hello world!" <<endl;
    cout<< "i am a student";
    return 0;
}
```

ماذا سيحدث لو لم يوضع endl في نهاية التعليمة الأولى؟

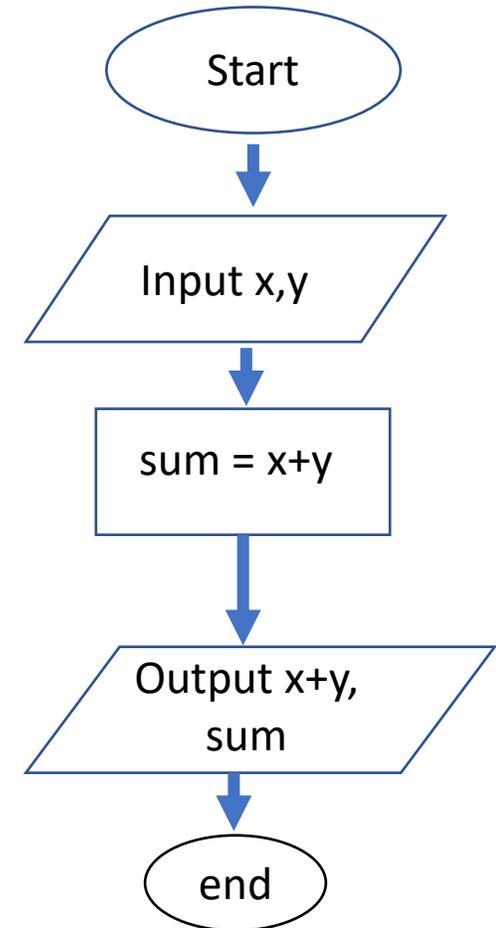
ليكن لدينا البرنامج التالي

```
#include <iostream>
using namespace std;

int main()
{int x,y,sum;// int x ; int y ; int sum;
cin>>x>>y;// cin>>x; cin>>y;
sum=x+y;
cout<<x+y<<endl<<sum;
return 0;
}
```

انتبه :

- x, y تم التصريح عنهما كتحويلين من نمط صحيح
- لاحظ استخدام // لإضافة التعليقات
- تم ادخال القيم لكل من x, y
- نلاحظ حساب قيمة المجموع واسنادها الى sum
- تم طباعة القيمة المحسوبة $x*y$ والنزول سطر ثم طباعة قيمة المجموع



ليكن لدينا البرنامج التالي

```
#include <iostream>
using namespace std;

int main()
{int x,y,sum;// int x ; int y ; int sum;
cin>>x>>y;// cin>>x; cin>>y;
sum=x+y;
cout<<x+y<<endl<<sum;
return 0;
}
```

انتبه :

- لو جعلنا تعليمات الإدخال تعليق لاحظ أن المتحولين x, y يأخذان قيم عشوائية من الذاكرة
- لو جعلنا تعليمات التصريح تعليق لاحظ ان المترجم compiler سيعطي خطأ لأن كل من x, y غير معرف في هذا المجال
- نستطيع طباعة القيمة $x+y$ مباشرة دون تخزينها في متحول

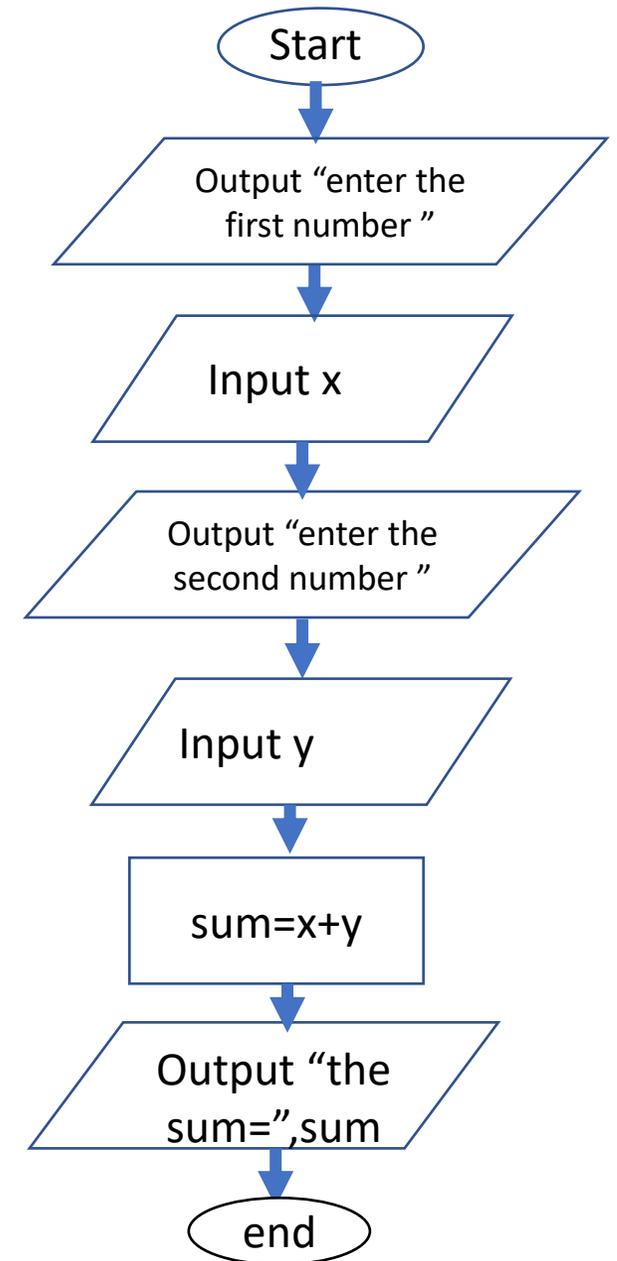
ليكن لدينا البرنامج التالي

```
#include <iostream>
using namespace std;

int main()
{int x,y,sum;// int x ; int y ; int sum;
cout <<"enter the first number";
cin>>x;
cout <<"enter the second number";
cin>>y;
sum=x+y;
cout <<"the sum =" <<sum;
}
```

انتبه :

- يمكننا أن نرتب الخرج
من خلال رسائل
توضيحية كما هو مبين
في الكود المجاور



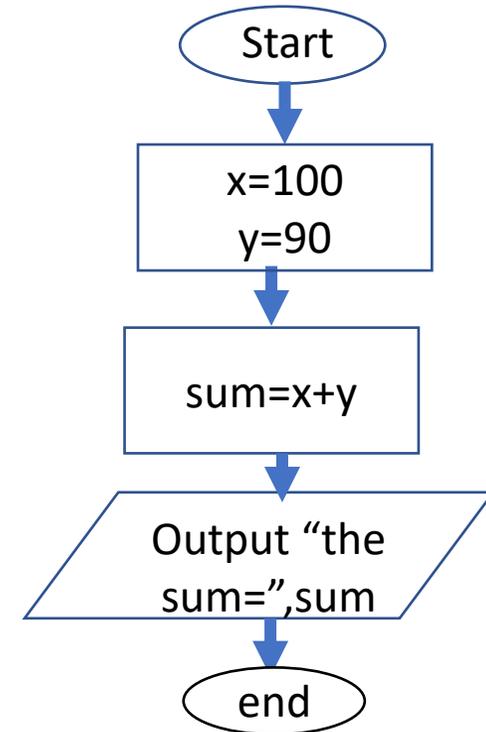
ليكن لدينا البرنامج التالي

```
#include <iostream>
using namespace std;

int main()
{int x,y,sum;// int x ; int y ; int sum;
/*cout <<"enter the first number";
cin>>x;
cout <<"enter the second number";
cin>>y;*/
x=100;
y=90;
sum=x+y;
cout <<"the sum =" <<sum;
}
```

انتبه :

- يمكن أن نجعل كتلة من التعليمات تعليق باستخدام /* */
- لاحظ أننا قمنا بإسناد القيم إلى x, y



ليكن لدينا البرنامج التالي

```
#include <iostream>
using namespace std;

int main()
{int x=100 ;
int y=90;
int sum;
sum=x+y;
cout <<"the sum =" <<sum;
}
```

انتبه :

- ونستطيع اسناد القيم بهذا الشكل
ملاحظة هامة :
ان لغة ++c حساسة لحالة الأحرف

انتهت تمارين الأسبوع ٥

جامعة
المنارة
HIMARA UNIVERSITY