

# نخستین برنامه

مبانی برنامه نویسی

(۱۱-۱۳-۱۳۹۱)

جلسه هفتم



دانشگاه شهید بهشتی

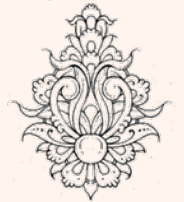
پاییز ۱۳۹۱

دانشکده مهندسی برق و کامپیوتر

احمد محمودی ازناوه

# فهرست مطالب

- سطوح برنامه‌نویسی
- مقایسه‌ی مختصر زبان‌های برنامه‌نویسی
- تاریخچه‌ی C++
- نخستین برنامه



# سطوح برنامه‌نویسی

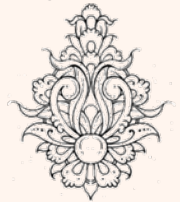
• زبان‌های برنامه‌نویسی به سه گروه عمده تقسیم می‌شوند:

– زبان‌های سطح بالا مانند: Pascal, Basic, Cobol, Fortran

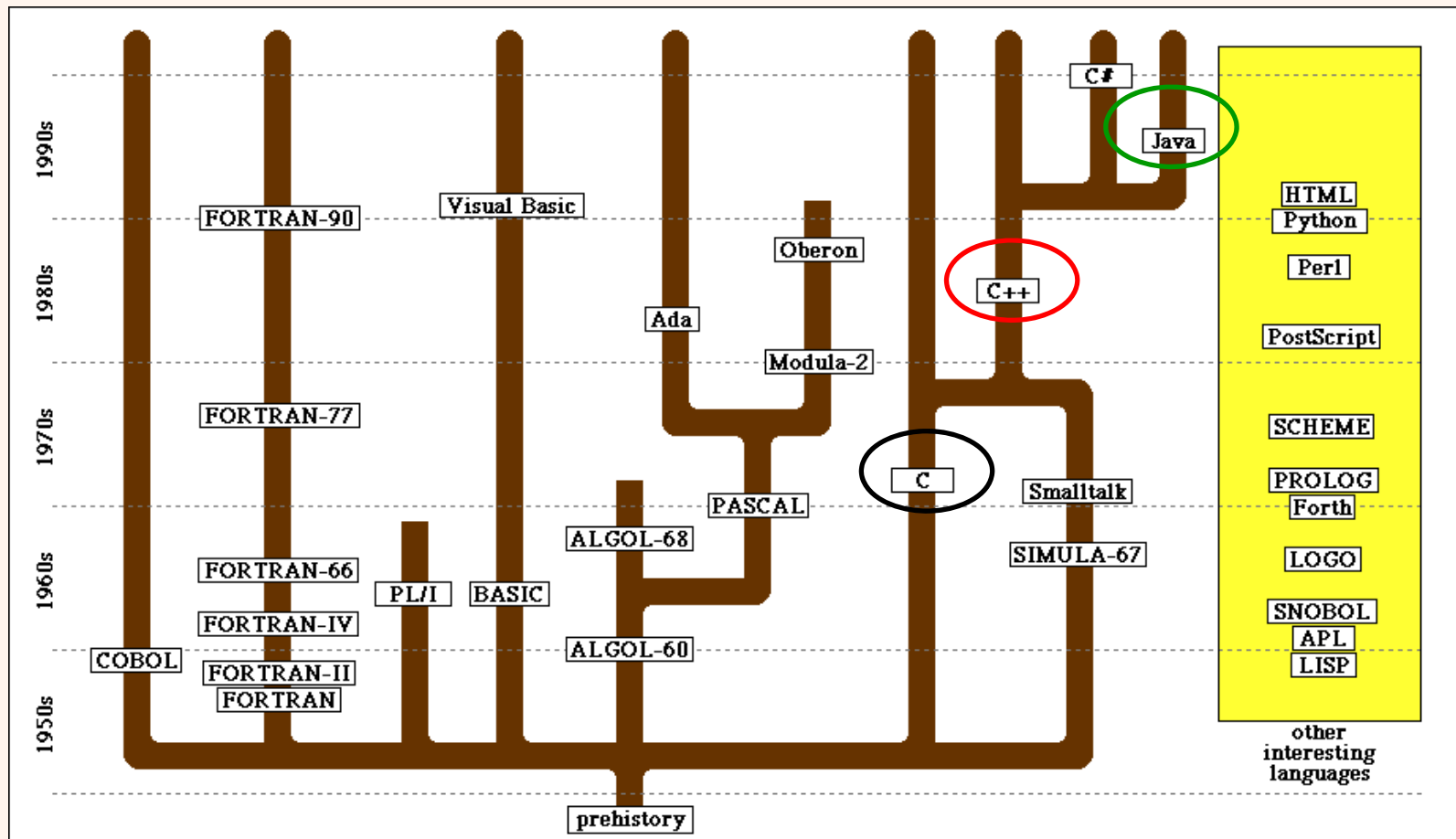
– زبان‌های میانه مانند: C و C++

– زبان‌های سطح پایین: اسمبلی و زبان ماشین

• سطح میانه بودن بدین معناست که این زبان امکانات و قدرت زبان‌های سطح پایین را داراست و همچنین عناصر زبان‌های سطح بالا را نیز پشتیبانی می‌کند.



# سیر تکاملی زبان‌های برنامه‌نویسی



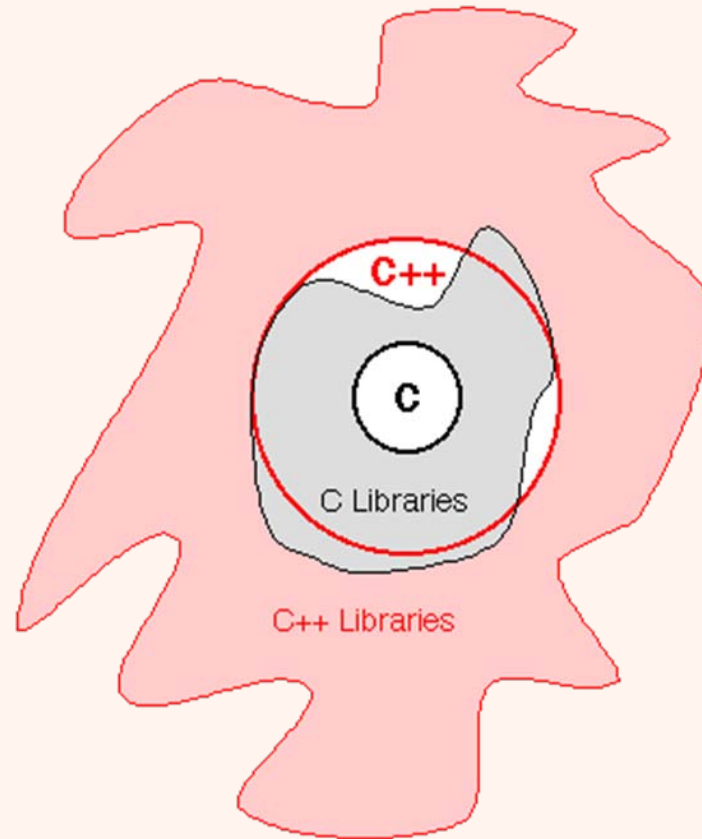
برگرفته از اسلایدهای دانشگاه استنفورد

# بزرگی و پیچیدگی زبان‌های برنامه‌نویسی

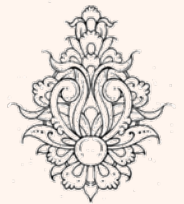


برگرفته از اسلایدهای دانشگاه استنفورد

# مقایسه میان زبان‌های C و C++



برگرفته از اسلایدهای دانشگاه استنفورد



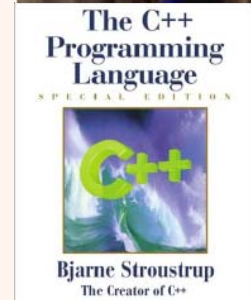
# تاریخچه C++

در دهه ۱۹۷۰ در آزمایشگاه‌های بل زبانی به نام C ایجاد شد.

انحصار این زبان در اختیار شرکت بل بود تا این که در سال ۱۹۷۸ توسط Kernighan و Richie شرح کاملی از این زبان منتشر شد و به سرعت نظر برنامه‌نویسان حرفه‌ای را جلب نمود.

هنگامی که بحث شی‌گرایی و مزایای آن در جهان نرودافزار رونق یافت، در اوایل دهه‌ی ۱۹۸۰ Bjarne Stroustrup زبان C++ را طراحی نمود.

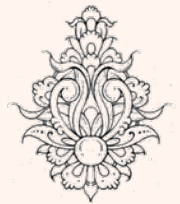
در سال ۱۹۹۸ زبان C++ توسط موسسه‌ی استانداردهای ملی آمریکا (ANSI) به شکل استاندارد و یک‌پارچه درآمد.



# برگرفته از کتاب **Programming Principle and Practice using c++**

# C++

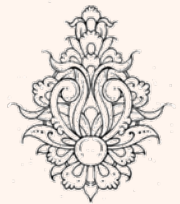
- در این زبان عملگرهایی تعبیه شده که برنامه نویسی سطح پایین و به زبان ماشین را نیز امکان پذیر می سازد.
- زبان **C** برای اجرای بسیاری از دستوراتش از توابع کتابخانه ای استفاده می کند و بیشتر خصوصیات وابسته به سخت افزار را به این توابع واگذار می نماید.
- برنامه ی مقصدی که توسط کامپایلرهای **C** ساخته می شود بسیار فشرده تر و کم حجم تر از برنامه های مشابه در سایر زبان ها است.
- **C++** که از نسل **C** است، تمام ویژگی های جذاب **C** را به ارث برده است .
- **C++** زبانی است که به صورت گسترده در محدودی برنامه های کاربردی و مهندسی استفاده می شود.
- برنامه نویسی به این زبان کاربر را قادر می سازد به زبان هایی چون **Java** و **C#** به آسانی مسلط گردد.





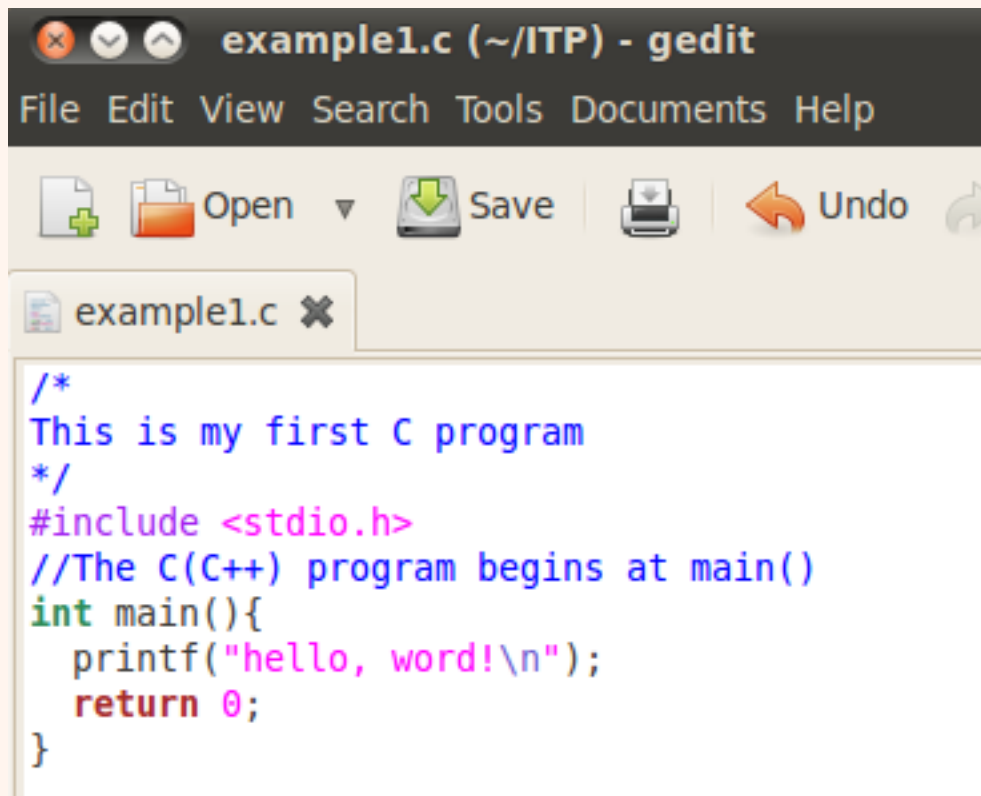
## شروع به کار

- یک «برنامه» دستورالعمل‌های متوالی است که می‌تواند توسط کامپیوتر اجرا شود.
- برای نوشتن و اجرای هر برنامه به یک «ویرایش‌گر متن» و یک «کامپایلر» احتیاج داریم.
- برای نوشتن و اجرا از ویرایش‌گر متن موجود در محیط لینوکس و برای کامپایل از **GCC** استفاده می‌کنیم.  
(**GNU Compiler Collection**)
- **Visual C++** محصول شرکت میکروسافت و **C++** **Builder** محصول شرکت بورلند نمونه‌های جالبی از محیط مجتمع تولید برای زبان **C++** به شمار می‌روند.

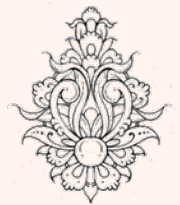


# نوشتن نخستین برنامه به زبان C

- تنها راه آموختن یک زبان نوشتن برنامه‌ای به آن زبان است، تقریباً اولین برنامه‌ای که به هر زبانی نوشته می‌شود، برنامه‌ی «**hello world!**» است؛ برنامه‌ای که عبارت گفته شده را چاپ کند.



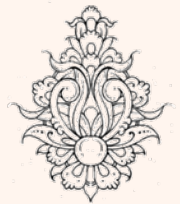
```
example1.c (~/ITP) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
example1.c
/*
This is my first C program
*/
#include <stdio.h>
//The C(C++) program begins at main()
int main(){
    printf("hello, word!\n");
    return 0;
}
```



# توضیحات برنامه

- خط نخست: **توضیح (comment)**، متنی است که به منظور راهنمایی و درک بهتر به برنامه اضافه می‌شود و تاثیری در اجرای برنامه ندارد. کامپایلر توضیحات برنامه را قبل از اجرا حذف می‌کند. این به معنای بی‌اهمیت بودن این توضیحات نیست. یک برنامه‌ی خوب، برنامه‌ای است که علاوه بر کارایی قابلیت فهم بالایی هم داشته باشد.

```
/*  
    This is my first C program  
*/  
#include <stdio.h>  
// The C(C++) program begins at main()  
int main(){  
    printf("hello, world!\n");  
    return 0;  
}
```



# توضیحات برنامه (ادامه ...)

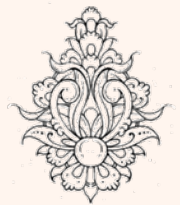
```
/*  
    This is my first C program  
*/  
#include <stdio.h>  
// The C(C++) program begins at main()  
int main(){  
    printf("hello, world!\n");  
    return 0;  
}
```

با استفاده از دو علامت اسلش «//» : هر متنی که بعد از دو علامت اسلش بیاید تا پایان همان سطر یک توضیح تلقی می‌شود .

هر متنی که با علامت «/\*» شروع شود و با علامت «\*/» پایان یابد یک توضیح تلقی می‌شود. این شیوه در استاندارد اولیه زبان C وجود نداشت، در زبان C++ و نهایتاً به زبان C هم اضافه شد.

```
//line comment
```

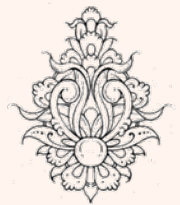
```
/*  
block comment  
*/
```



# راهنمای پیش پردازش

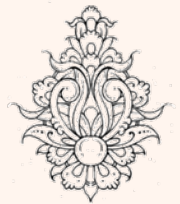
- دومین خط از کد بالا یک «راهنمای پیش پردازنده» یا همان **preprocessor** است. در واقع پیش پردازش گامی است که پیش از کامپایل برنامه انجام می شود.
- راهنمای پیش پردازنده با کاراکتر «#» شروع می شود، که نشان می دهد این خط، یک راهنمای پیش پردازنده است. این کاراکتر باید در ابتدای همی خطوط راهنمای پیش پردازنده باشد.
- عبارت `include` – در واقع به کامپایلر اعلام می کند که در این برنامه از توابع کتابخانه ای استفاده شده است.
- نام یک «فایل کتابخانه ای» میان دو علامت `<>` قرار می گیرد. به چنین فایلی فایل سرآیند می گویند.

```
/*  
  This is my first C program  
*/  
#include <stdio.h>  
// The C(C++) program begins at main()  
int main(){  
  printf("hello, world!\n");  
  return 0;  
}
```



# شروع برنامه

- در واقع برنامه‌ی ما با عبارت «**main**» آغاز می‌شود.
  - عبارت **int** که یک نوع عددی در C++ است. در واقع نوع خروجی تابع main را مشخص می‌کند.
  - اگر برنامه با موفقیت اجرا شود مقدار عددی صفر باز می‌گردد.
  - عبارت **main** که به آن «**تابع اصلی**» در C++ می‌گویند.
  - دو پرانتز ( ) که نشان می‌دهد عبارت main یک «تابع» است.
  - بدنه‌ی برنامه بین دو علامت { } قرار می‌گیرد.
  - تابع printf عبارت رشته‌ای محصور در "" را چاپ می‌کند، این تابع متعلق به کتابخانه‌ی stdio می‌باشد.

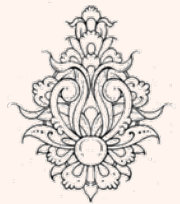


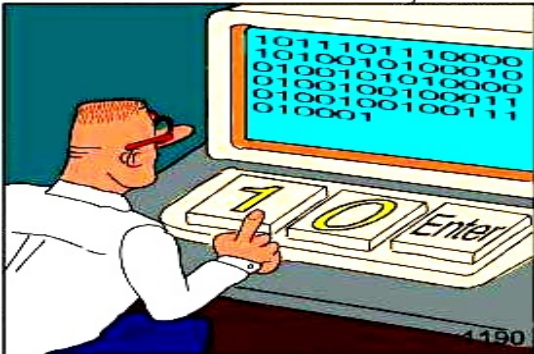
## دو شیوهی مختلف

- دو تصویر زیر از لحاظ کامپایلر هیچ تفاوتی ندارند. به نظر شما کدام شیوه مناسبتر است؟
- به طور کلی شیوهی نوشتن همراه با تورفتگی توصیه می‌شود، در آینده در این مورد بیشتر صحبت می‌کنیم.

```
// The C(C++) program begins at main()
int main(){
    printf("hello, world!\n");
    return 0;
}
```

```
int main(){ printf("hello, world!\n"); return 0; }
```



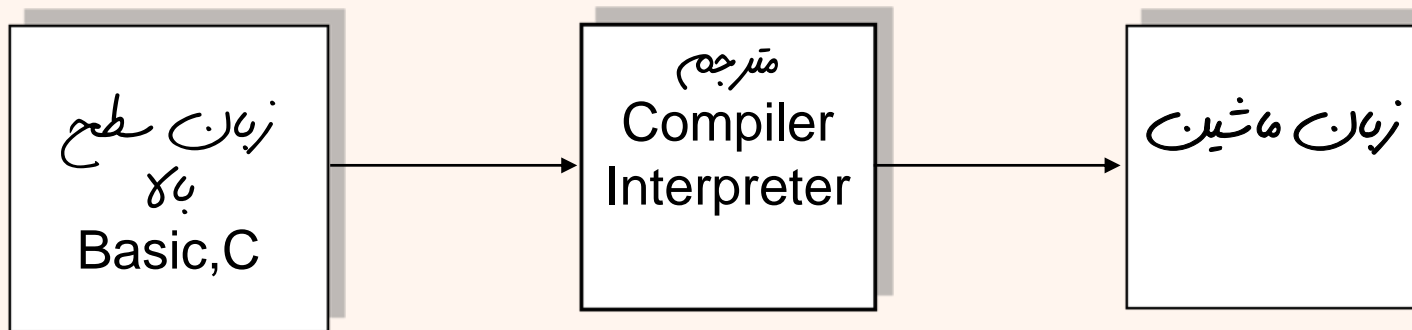


# ترجمه‌ی برنامه به زبان ماشین

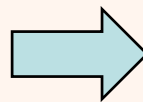
• انواع مترجم

COMPILER -

INTERPRETER -



```
/*  
 This is my first C program  
*/  
#include <stdio.h>  
// The C(C++) program begins at main()  
int main(){  
 printf("hello, world!\n");  
 return 0;  
}
```



```
00000000 <main>:  
0: 55          push   %ebp  
1: 89 e5      mov    %esp,%ebp  
3: 83 e4 f0   and   $0xfffffff0,%esp  
6: 83 ec 10   sub   $0x10,%esp  
9: c7 04 24 00 00 00 00  movl  $0x0,(%esp)  
10: e8 fc ff ff  call  11 <main+0x11>  
15: b8 00 00 00 00  mov   $0x0,%eax  
1a: c9        leave  %eax  
1b: c3        ret
```

