دوره مهندسی معکوس نرم افزار

- Site: OnHexGroup.ir
- Youtube: @onhexgroup
- Telegram: onhex_ir
- Twitter: @onhexgroup
- Github: onhexgroup

نسخه ی ۳۲ و ۶۴ بیتی

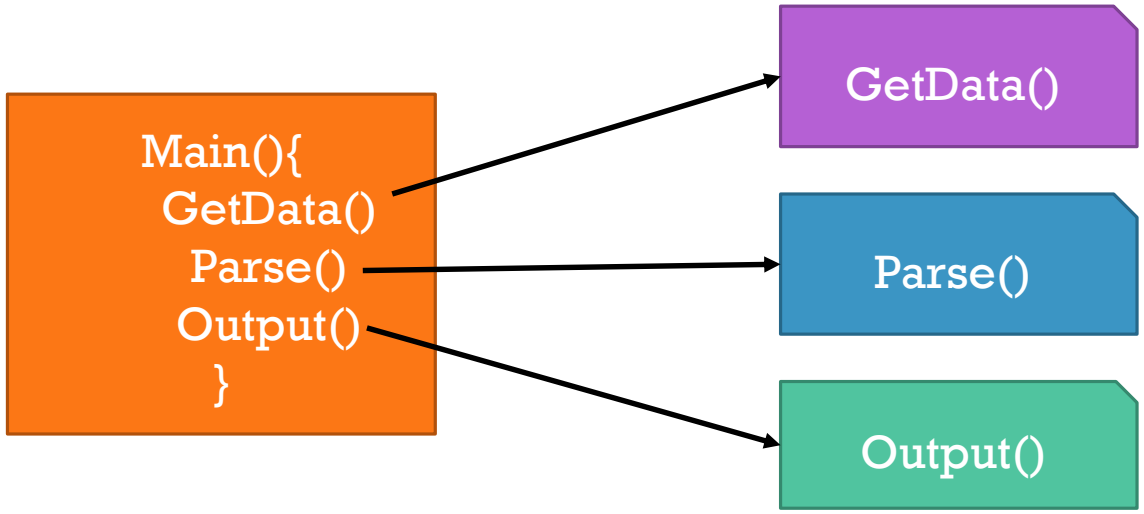
پلتفرم: ویندوز

ارائه دهنده : **OnhexGroup**

Onhexgroup.ir

توابع

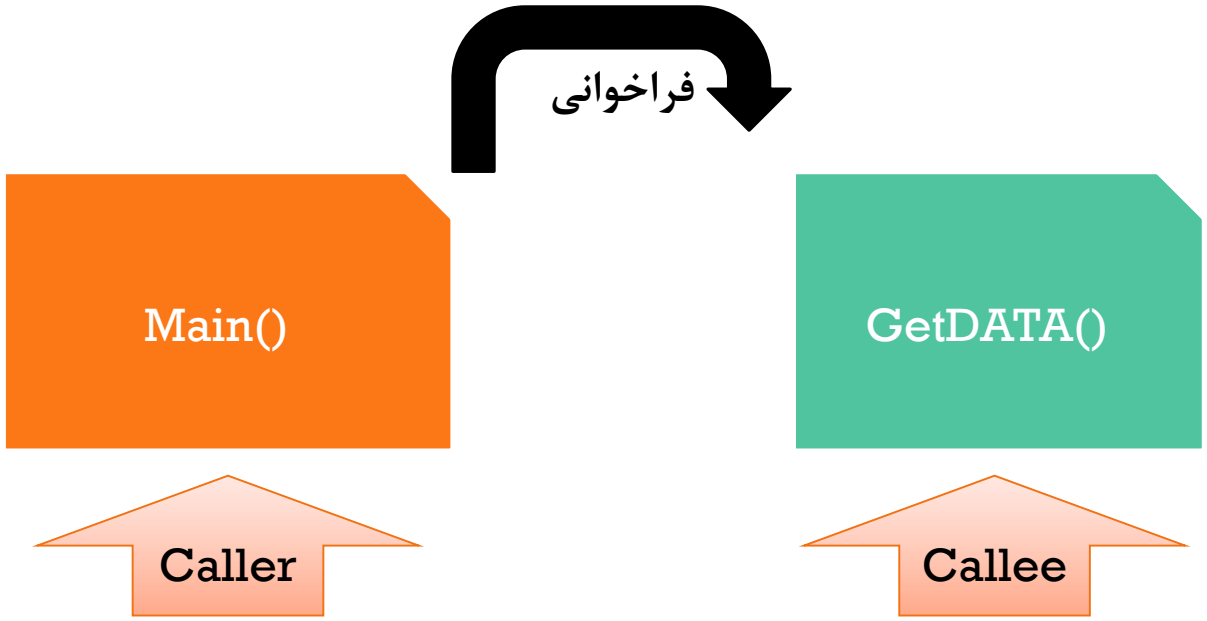
- به کمک توابع میتونیم برنامه های بزرگ رو به قسمت های کوچیک تقسیم کنیم:
  - توسعه برنامه ساده تر میشه
  - نگهداری و عیب یابی برنامه ساده تر میشه.
- تابع، روال ، رویه، متد، زیربرنامه، **Subroutine** ، **Procedure**



Twitter:Onhexgroup

کار با توابع

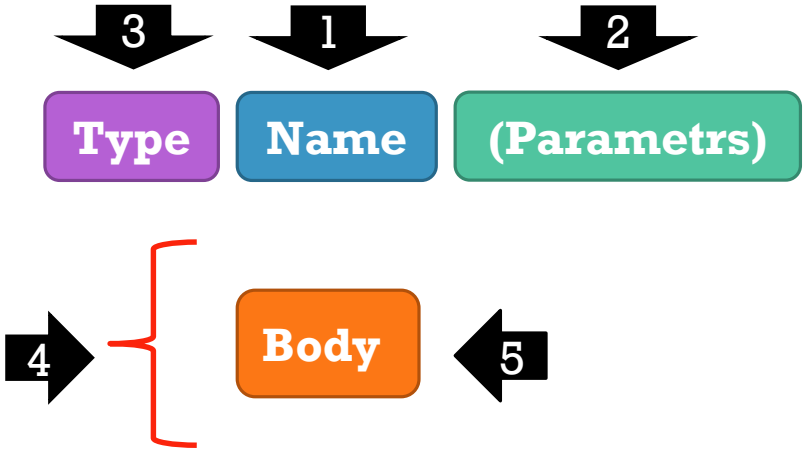
- برای کار با توابع:
  - نحوه ی تعریف یک تابع
  - نحوه ی فراخوانی یک تابع



Twitter:Onhexgroup

موارد تشکیل دهنده ی تابع

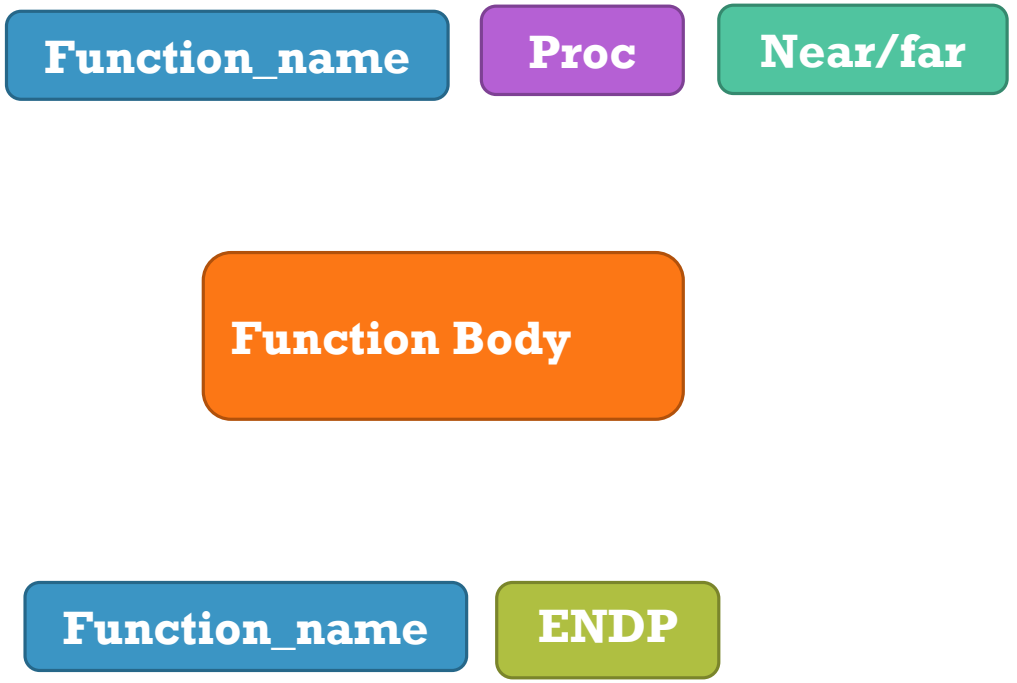
- یک تابع میتونه از موارد زیر تشکیل بشه:
  1. نام
  2. پارامترهای ورودی
  3. نوع
  4. محدوده
  5. بدنه



- در اسمبلی برای تعریف توابع از ساختار زیر استفاده میکنیم:

Github: Onhexgroup

تعریف توابع



- در C/C++ برای تعریف توابع از ساختار زیر استفاده میکنیم:
(Inline assembly 32 bit)

Github: Onhexgroup

تعریف توابع

Onhexgroup.ir

فراخوانی توابع

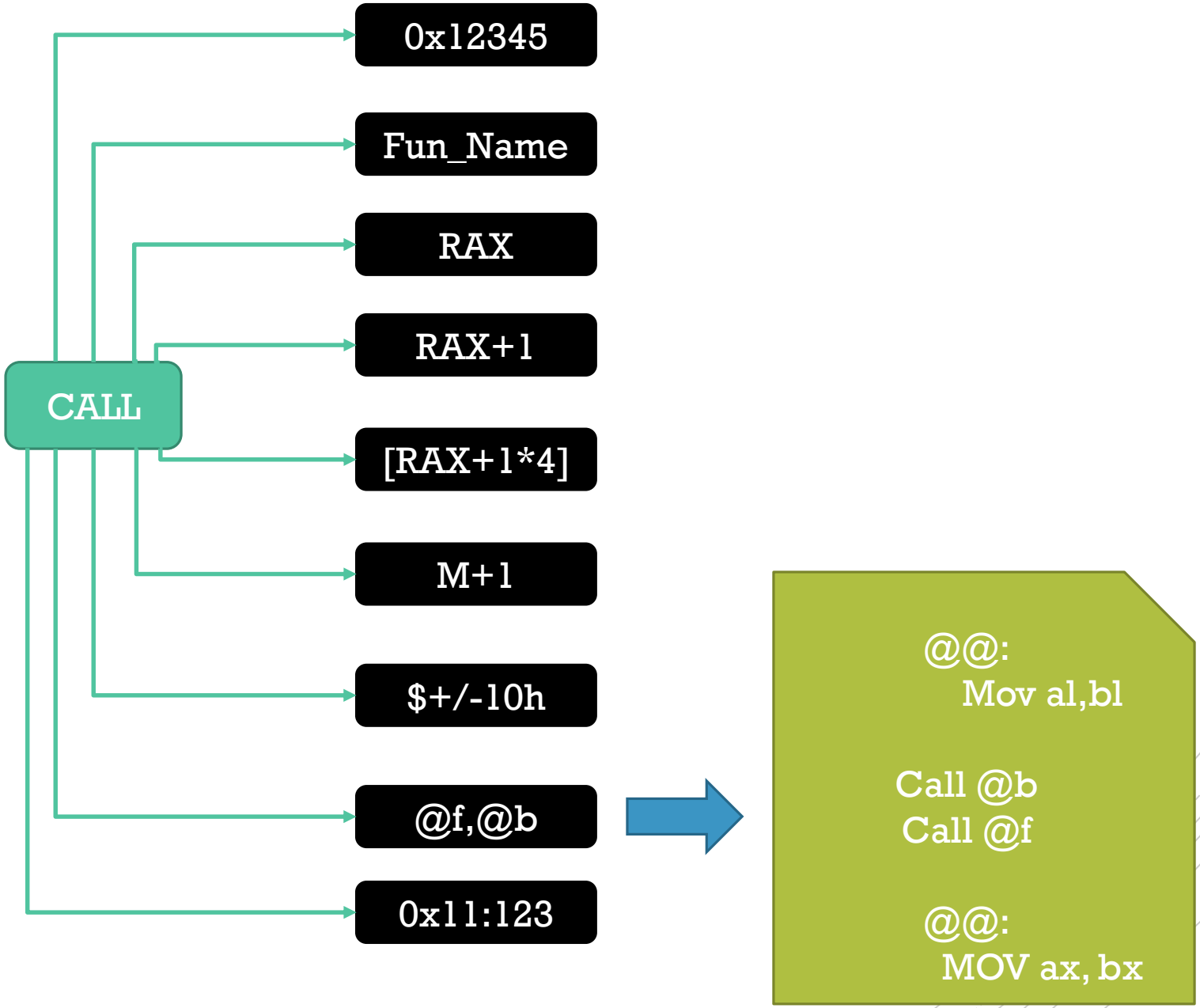
- برای فراخوانی توابع از دستور **Call** استفاده میکنیم.
- عملکرد این دستور:
  - آدرس بازگشت را در پشته قرار میدهد
  - اجرای برنامه رو به تابع فراخوانی شده، میبره. (سر تابع)

CALL R/M/IMM

- انواع دستور **CALL**:
  - **NEAR** یا نزدیک : در همان سگمنت کد
  - **FAR** یا دور: در یک سگمنت دیگه
  - **Inter-privilege-level far** : همان نوع **FAR** فقط سطح دسترسی فرق داره.
  - **Task switch**: در یک تسک دیگه (در ۶۴ بیتی پشتیبانی نمیشه)

Onhexgroup.ir

فراخوانی توابع



Onhexgroup.ir

بازگشت از توابع

- برای بازگشت از توابع از دستور **RET** استفاده میکنیم.
- عملکرد این دستور:
  - آدرس بازگشت رو از پشته در **EIP/RIP** قرار میده
  - اجرای برنامه رو به آدرس مورد نظر، میبره. (آدرس بازگشت)

RET IMM16

- انواع دستور **RET**:
  - **NEAR** یا نزدیک : در همان سگمنت کد
  - **FAR** یا دور: در یک سگمنت دیگه
  - **Inter-privilege-level far** : همان نوع **FAR** فقط سطح دسترسی فرق داره.