



بخش

## مبانی برنامه‌نویسی

شامل فصل‌های

- فصل ۱ معرفی زبان C#
- فصل ۲ مقدمات زبان C#
- فصل ۳ آشنایی با مفاهیم برنامه‌نویسی  
ویژوال



# معرفی زبان C#

## مقدمه

کامپیوتر از دو جزء اصلی **سخت افزار** (Hardware) و **نرم افزار** (Software) تشکیل شده است. نرم افزار مجموعه‌ای از یک یا چند **برنامه کامپیوتری** (Computer Program) است که با استفاده از دستورات و قواعد یک زبان برنامه‌نویسی توسط برنامه‌نویس نوشته می‌شود و سخت‌افزار را قابل استفاده می‌کند. هدف از هر نرم‌افزاری، پیاده‌سازی یک **الگوریتم** (Algorithm) بر روی یک ماشین (همانند کامپیوتر، تلفن همراه، قطعه‌ای الکترونیکی و غیره) و بهره‌گیری از منابع سخت‌افزاری و نرم‌افزاری آن است. در این فصل ابتدا با مفهوم الگوریتم آشنا شده و در نهایت ویژگی‌ها و اصطلاحات مرتبط با زبان C# را فرا خواهید گرفت. به طور کلی پس از اتمام این فصل قادر خواهید بود:

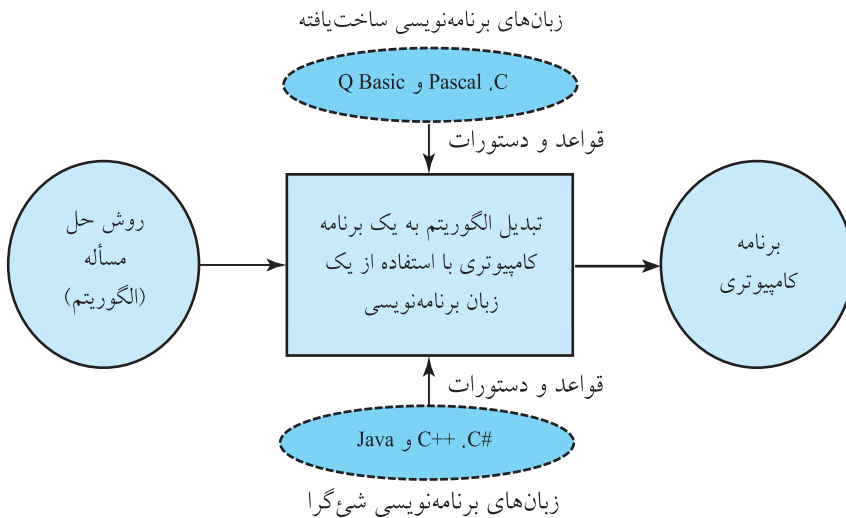
- سبک‌های برنامه‌نویسی ساخت‌یافته و شیء‌گرا را شرح دهید.
- چند مورد از مزایای زبان C# را نسبت به سایر زبان‌های برنامه‌نویسی نام ببرید.
- مفهوم چارچوب دات‌نت را شرح دهید و همچنین اجزای آن را نام ببرید.
- نحوه کامپایل یک برنامه را در چارچوب دات‌نت شرح دهید.

## الگوریتم و برنامه کامپیوتری

واژه الگوریتم از نام «ابوجعفر محمدبن موسی الخوارزمی» ریاضیدان و منجم ایرانی در قرن دوم هجری گرفته شده است و به معنی تشریح دقیق مراحل و نحوه انجام دادن کار به خصوصی است و به صورت زیر تعریف می‌شود:

«به مجموعه‌ای از دستورات که با زبان دقیق و قابل فهم به همراه جزییات لازم و به صورت مرحله به مرحله به گونه‌ای اجرا شده که **هدف خاصی** (حل مسأله) را دنبال کنند و شروع و خاتمه آن‌ها نیز مشخص باشد، الگوریتم می‌گویند.»

همان‌طور که بیان شد، بین **نرم‌افزار** (برنامه کامپیوتری) و الگوریتم، ارتباط مستقیمی وجود دارد. بیان الگوریتم براساس دستورات یک زبان برنامه‌نویسی را یک برنامه کامپیوتری در آن زبان می‌نامند. به عنوان مثال، اگر الگوریتمی را براساس قواعد و دستورات زبان C# پیاده‌سازی کنید، از آن الگوریتم، یک برنامه کامپیوتری به زبان C# به دست می‌آید. همان‌طور که در شکل ۱-۱ نشان داده شده است، الگوریتم با استفاده از دستورات یک زبان برنامه‌نویسی ساخت‌یافته (مثل زبان C) و یا یک زبان برنامه‌نویسی شی‌گرا (مثل زبان C++) به یک برنامه کامپیوتری تبدیل می‌شود.



شکل ۱-۱ تبدیل یک الگوریتم به یک برنامه کامپیوتری

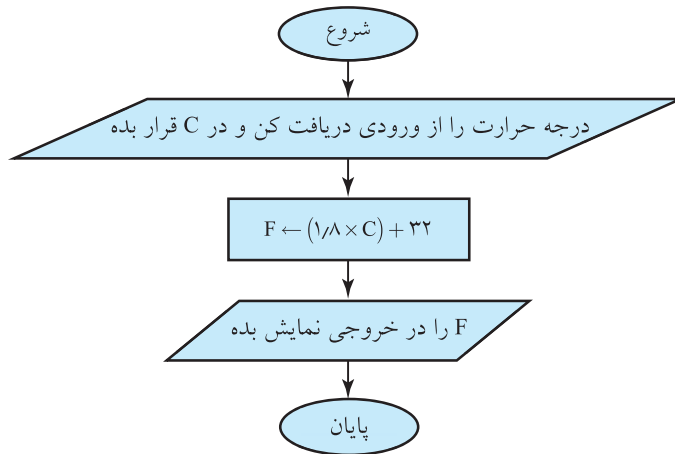
زبان‌های برنامه‌سازی همانند زبان‌های طبیعی تنوع زیادی دارند، به طوری که تقریباً برای هر نوع کاری یک یا چند زبان برنامه‌سازی طراحی شده است.

## روش‌های بیان الگوریتم

الگوریتم‌ها را می‌توان به روش‌های مختلفی بیان کرد که عبارت‌اند از:

۱. **روش ریاضی:** در این روش از علائم ریاضی برای بیان الگوریتم استفاده می‌شود. به عنوان مثال، الگوریتم زیر، سه عدد را از ورودی خواننده و میانگین آن‌ها را محاسبه می‌کند. روش بیان این الگوریتم، روش ریاضی است.

مفروضات	الگوریتم
عدد اول: X	۱. شروع
عدد دوم: Y	۲. X, Y و Z را از ورودی دریافت کن.
عدد سوم: Z	۳. $Sum \leftarrow X + Y + Z$
مجموع اعداد: Sum	۴. $Ave \leftarrow Sum \div 3$
میانگین اعداد: Ave	۵. Ave را نمایش بده.
	۶. پایان



شکل ۲-۱ فلوجارت تبدیل دمای سلسیوس به فارنهایت

۲. روش شبه کد (Pseudo Code): در این روش از ساختارهای معمول برخی از زبان‌های برنامه‌نویسی، زبان محاوره معمولی و علائم ریاضی برای بیان الگوریتم استفاده می‌شود. به عنوان مثال، الگوریتم زیر، یک مقدار را برحسب کیلومتر از ورودی خوانده و به متر تبدیل می‌کند. روش بیان این الگوریتم، روش شبه کد است.

```

start
input K
M := K * 1000
print M
end
  
```

۳. روش روندنما یا فلوجارت (Flowchart): در این روش از اشکال و نمادهای خاصی برای بیان الگوریتم استفاده می‌شود که در درس مبانی کامپیوتر با آن‌ها آشنا شده‌اید. به عنوان مثال، الگوریتم شکل ۲-۱، درجه حرارت را برحسب سلسیوس از ورودی خوانده و به فارنهایت تبدیل می‌کند. روش بیان این الگوریتم، روش فلوجارت است. در ادامه، با ارائه چند تمرین، با نحوه نوشتن الگوریتم برای مسائل مختلف آشنا خواهید شد.

**تمرین ۱-۱** الگوریتمی بنویسید که زوج یا فرد بودن یک عدد صحیح دلخواه را مشخص کند.

**راهنمایی:** در ریاضیات، به هر عدد صحیحی که باقیمانده تقسیم آن بر عدد ۲، برابر با صفر باشد، عدد زوج و در غیر این صورت عدد فرد گویند.

**پاسخ** الگوریتم مسأله فوق به صورت زیر است:

بیان الگوریتم به روش شبه کد	بیان الگوریتم به روش ریاضی
start	۱. شروع
input X	۲. یک عدد از ورودی دریافت کن و آن را در X قرار بده.
R:= X mod 2	۳. $R \leftarrow X \bmod 2$
if R = 0 then	۴. اگر $R = 0$ بود، آنگاه «عدد ورودی زوج است» را نمایش بده.
print " عدد ورودی زوج است."	۵. اگر $R = 1$ بود، آنگاه «عدد ورودی فرد است» را نمایش بده.
else	۶. پایان
print " عدد ورودی فرد است."	
end	

**یادآوری:** منظور از  $mod$  در بیان الگوریتم، باقیمانده تقسیم صحیح است. به عنوان مثال، باقیمانده تقسیم ۷ بر ۲ برابر با ۱ است:

$$7 \bmod 2 \rightarrow 1$$

**تمرین ۱-۲** الگوریتمی بنویسید که مقسوم علیه‌های عدد طبیعی و دلخواه M را محاسبه کند.

**راهنمایی:** هر عدد طبیعی مثل X بر تعدادی از عددها بخش پذیر است، به این عددها مقسوم علیه‌های عدد X می‌گویند. به عنوان مثال، عدد ۲۰ بر عددهای ۱، ۲، ۴، ۵، ۱۰ و ۲۰ بخش پذیر است. بنابراین خواهیم داشت:

$$\{1, 2, 4, 5, 10, 20\} = \text{مجموعه مقسوم علیه‌های عدد } 20$$

**پاسخ** الگوریتم مسأله فوق به صورت زیر است:

بیان الگوریتم به روش شبه کد	بیان الگوریتم به روش ریاضی
start	۱. شروع
input M	۲. یک عدد از ورودی دریافت کن و آن را در M قرار بده.
K = 1	۳. $K \leftarrow 1$
L1:	۴. اگر $M \bmod K = 0$ بود، آنگاه K را نمایش بده.
if M mod K = 0 then	۵. $K \leftarrow K + 1$
print K	۶. اگر $K \leq M$ بود، آنگاه برو به مرحله ۴.
K = K + 1	
if K <= M then	
go to L1:	
end	

**تمرین ۱-۳** الگوریتمی بنویسید که قیمت یک کالا را از ورودی خوانده و قیمت نهایی آن را با احتساب ۱۵ درصد تخفیف محاسبه و نمایش دهد.

**پاسخ** الگوریتم مسأله فوق به صورت زیر است:

بیان الگوریتم به روش شبه کد	بیان الگوریتم به روش ریاضی
start	۱. شروع
input N	۲. قیمت کالا را دریافت کن و آن را در N قرار بده.
M = N - (N * 0.15)	۳. $M \leftarrow N - (N \times 0.15)$
print M	۴. M را نمایش بده.
end	۵. پایان

### تمرین ۱-۴ الگوریتمی بنویسید که انرژی جنبشی یک جسم را برحسب ژول محاسبه کند.

**راهنمایی:** انرژی‌ای که هر جسم، صرفاً به علت حرکتش دارد، انرژی جنبشی آن جسم نامیده می‌شود. انرژی جنبشی جسمی به جرم  $M$  که با سرعت  $V$  حرکت می‌کند، از رابطه زیر محاسبه می‌شود:

$$K = \frac{1}{2} \times M \times V^2$$

در این رابطه، جرم برحسب کیلوگرم، سرعت برحسب متر بر ثانیه و انرژی جنبشی برحسب ژول است.

**پاسخ** الگوریتم مسأله فوق به صورت زیر است:

بیان الگوریتم به روش شبه کد	بیان الگوریتم به روش ریاضی
start	۱. شروع
input M	۲. جرم جسم را برحسب کیلوگرم دریافت کن و آن را در $M$ قرار بده.
input V	۳. سرعت جسم را برحسب متر بر ثانیه دریافت کن و آن را در $V$ قرار بده.
$K = (0.5 * M) * (V * V)$	۴. $K \leftarrow (0.5 \times M) \times (V \times V)$
print K	۵. $K$ را نمایش بده.
end	۶. پایان

## سبک‌های برنامه‌نویسی

همان‌طور که در شکل ۱-۱ نشان داده شده است، دو سبک برنامه‌نویسی رایج که برنامه‌نویسان از آنها برای توسعه سیستم‌های نرم‌افزاری استفاده می‌کنند عبارت‌اند از:

۱. **برنامه‌نویسی ساخت‌یافته (Structured Programming):** در این سبک برنامه‌نویسی، برنامه به صورت مجموعه‌ای از فعالیت‌ها در نظر گرفته می‌شود که باید بر روی داده‌ها اجرا شوند. هر کار پیچیده‌ای به چند کار کوچک‌تر تجزیه می‌شود تا این کارها به راحتی قابل پیاده‌سازی باشند. سپس برای انجام هر یک از این کارها، زیر برنامه خاصی (که **متد** (Method) نامیده می‌شود) نوشته می‌شود. برنامه‌نویسی ساخت‌یافته، روش مناسبی برای حل مسائل پیچیده است، اما مشکلات خاص خودش را دارد. در این روش داده‌ها از **فعالیت‌ها** (متدها) که آنها را پردازش می‌کنند جدا هستند. وقتی حجم داده‌ها زیاد می‌شود، نگهداری آنها مشکل می‌شود. هر چه بیشتر با داده‌ها کار می‌کنید، با مشکلات بیشتر تری مواجه می‌شوید.

متدها قابلیت استفاده مجدد را دارند. یعنی می‌توان متدهای یک برنامه را در برنامه‌های دیگر نیز استفاده کرد. به این ترتیب، کدنویسی کم‌تر می‌شود و توسعه نرم‌افزار با سرعت بیشتری صورت می‌پذیرد. متدها این امکان را فراهم می‌کنند تا توسعه نرم‌افزار با استفاده از گروه ممکن شود. به این شکل که پس از مشخص شدن متدهای مورد نیاز برنامه، هر یک از اعضای گروه وظیفه نوشتن و تست متدهای مشخصی را برعهده می‌گیرد و به این شکل اعضای گروه به صورت همزمان روی بخش‌های مختلف برنامه فعالیت می‌کنند بدون اینکه یکی منتظر دیگری بماند. در نهایت این متدها به هم پیوند می‌خورند تا جریان منطقی برنامه برقرار شود. بدیهی است که عیب‌یابی و اصلاح برنامه نیز آسان می‌شود. زیرا، با بررسی جریان برنامه، متد معیوب به سرعت شناسایی و اصلاح می‌شود بدون اینکه بقیه بخش‌های برنامه تغییر یابند. زبان‌های C و Pascal دو نمونه از زبان‌های برنامه‌نویسی ساخت‌یافته هستند.

۲. **برنامه‌نویسی شیء‌گرا (Object Oriented Programming):** برنامه‌نویسی شیء‌گرا، جدیدترین سبک برنامه‌نویسی است که در آن همه چیز به دید **شیء** (Object) نگریسته می‌شود. شیء از یک **کلاس** (Class)

مشتق می‌شود. برنامه‌نویسان با شناسایی مسأله و انتزاع **موجودیت‌های** (Entities) آن، کلاس‌ها را طراحی می‌کنند.

هر کلاس مجموعه‌ای از **ثابت‌ها** (Constants)، **خاصیت‌ها** (Properties)، متدها و غیره است که یک بار طراحی می‌شود، اما می‌توان چندین **نمونه** (Instance) از آن ایجاد کرد. در حال حاضر پرطرفدارترین سبک برنامه‌نویسی، این سبک است.

ایده اصلی در زبان‌های شی‌گرا ترکیب داده‌ها و متدها در یک موجودیت واحد تحت عنوان شی است که در آن متدها عملیات مورد نظر را بر روی داده‌ها انجام می‌دهند. زبان‌های شی‌گرا با الهام و تبعیت از دنیای واقعی به وجود آمده‌اند بدین گونه که در دنیای واقعی به هر جا که بنگریم، موارد زیادی از نمونه‌های متمایز و متفاوت و در عین حال با ساختار و قالبی یکسان قابل مشاهده است. به عنوان مثال، تمامی انسان‌ها با تفاوت‌های بسیار آفریده شده‌اند، به گونه‌ای که هیچ دو انسانی ویژگی‌های کاملاً یکسان ندارند (مثل تفاوت در رنگ چشم‌ها). با این حال همگی دارای ساختار و قالب یکسانی هستند (مثل ساختار سیستم دفاعی بدن). به این ساختار و قالب مشترک، کلاس گفته می‌شود و نمونه‌های موجود در یک ساختار را شی آن کلاس می‌نامند. به عنوان مثال، کلاس انسان را می‌توان ترکیبی از موارد زیر دانست:

- **خاصیت‌ها:** رنگ چشم، رنگ پوست، جنسیت، سن، قد، وزن و غیره
- **متدها:** راه رفتن، حرف زدن، دویدن و غیره
- **رویدادها:** عصبانی شدن، شاد شدن و غیره

پس از طراحی کلاس، با ایجاد نمونه‌هایی از آن، می‌توانید اشیای کاملاً متمایزی در اختیار داشته باشید. همانگونه که در دنیای ما، انسان‌های متمایز با ساختار یکسان ولی با رنگ چشم و رنگ پوست‌های متفاوت قابل مشاهده‌اند.

کلاس‌ها نسبت به اشیای مثل نقشه‌ها نسبت به ساختمان‌ها هستند، یعنی همان‌طور که از یک نقشه می‌توان چند ساختمان تهیه کرد، از یک کلاس می‌توان چند شی ایجاد کرد. یعنی، نقشه الگویی برای ساخت ساختمان و استفاده از آن است و کلاس الگویی برای ساخت اشیای و استفاده از آن‌ها است.

توسعه نرم‌افزار با استفاده از کلاس‌ها اجازه می‌دهد که سیستم‌های نرم‌افزاری آینده از کلاس‌های از قبل نوشته شده مجدداً استفاده کنند. به عنوان مثال، همان‌طور که تلویزیون را می‌توان با استفاده از قطعات آن ایجاد کرد، در آینده، نرم‌افزارها را نیز می‌توان با استفاده از کلاس‌های از قبل نوشته شده توسعه داد.

با بررسی سه اصل **نهم‌سازی** (Encapsulation)، **وراثت** (Inheritance) و **چندریختی** (Polymorphism) که از اصول مهم برنامه‌نویسی شی‌گرا محسوب می‌شوند، تفاوت برنامه‌نویسی شی‌گرا با برنامه‌نویسی ساخت‌یافته نمایان خواهد شد:

- **نهم‌سازی:** اشیای بدون اطلاع از چگونگی کارکرد یکدیگر می‌توانند با هم در ارتباط باشند.
- **وراثت:** یک کلاس می‌تواند رفتار یا صفاتی را از کلاس دیگر به ارث ببرد.
- **چندریختی:** اشیای می‌توانند در موقعیت‌های مختلف، رفتارهای متفاوتی از خود نشان دهند.

در فصل هفتم، مفاهیم برنامه‌نویسی شی‌گرا (ایجاد کلاس و شی) آموزش داده می‌شود. زبان‌های ++C، Java و #C از جمله زبان‌های برنامه‌نویسی شی‌گرا محسوب می‌شوند.

## نرم‌افزارهای کاربردی

نرم‌افزارها انواع مختلفی دارند. به عنوان مثال، سیستم عامل ویندوز، یک نرم‌افزار سیستمی و نرم‌افزار حسابداری هلو یک نرم‌افزار کاربردی است. هدف از نگارش این کتاب آموزش برنامه‌نویسی **نرم‌افزارهای کاربردی** (Ap-)

نرم افزارهای کاربردی نوعی از نرم افزارها هستند که در آنها ورود داده ها و خروج اطلاعات، بیش تر از محاسبات در واحد پردازشگر مرکزی (Central Processing Unit) است و معمولاً از پایگاه های داده (Data Bases) در آنها استفاده می شود. به عنوان چند نمونه از نرم افزارهای کاربردی می توان به موارد زیر اشاره کرد:

- سیستم حسابداری یک سازمان
- سیستم اتوماسیون یک سازمان
- سیستم مدیریت کتابخانه یک دانشگاه
- سیستم مدیریت یک هتل

زبان های C#, Delphi, Visual C++ و Java از جمله زبان های قدرتمندی هستند که می توان برای توسعه نرم افزارهای کاربردی از آنها استفاده کرد. در این کتاب برای توسعه نرم افزارهای کاربردی از زبان C# استفاده شده است. زبان C# به صورت اختصاصی برای کار با چارچوب دات نت (NET Framework) طراحی شده است. در ادامه، ضمن معرفی زبان C#، اصطلاحات مرتبط با این زبان نیز بررسی می شود.

## چارچوب دات نت

چارچوب دات نت یک فناوری نرم افزاری است که توسط شرکت مایکروسافت (Microsoft) برای سهولت در طراحی و توسعه نرم افزار ایجاد شده است و حاوی مجموعه ای از کلاس های کاربردی و مفید است. کلاس های موجود در این چارچوب بخش وسیعی از نیازهای برنامه نویسی را تحت پوشش قرار می دهند. چارچوب دات نت یک چارچوب کلی است و محدود به زبان برنامه نویسی خاصی نیست. مهم ترین زبان های برنامه نویسی برای کار در این چارچوب عبارت اند از:

- Visual C#
- Visual Basic
- Visual C++
- Visual F#

در چارچوب دات نت، یک برنامه که به زبان C# نوشته شده است به راحتی می تواند از کدی استفاده کند که به زبان Visual Basic نوشته شده است و بالعکس (ویژگی چند زبانی). تغییر عمده ای که در برنامه نویسی با چارچوب دات نت نسبت به گذشته قابل مشاهده است، نحوه اتصال به پایگاه داده است که می توان از روش جدید ADO.NET استفاده کرد. در این روش، نحوه اتصال و نوشتن داده ها کاملاً متفاوت با روش قدیمی یعنی ADO است که این تفاوت، تسهیل در برنامه نویسی و سرعت در تبادل داده ها را به ارمغان می آورد. در مورد نوع برنامه هایی که می توان با چارچوب دات نت نوشت هیچ محدودیت قابل ذکری وجود ندارد. می توانید برای طراحی برنامه های کاربردی تحت ویندوز، برنامه های تحت وب، برنامه های تحت کنسول (Con-sole) و غیره از این چارچوب استفاده کنید.

## اجزای چارچوب دات نت

چارچوب دات نت از چهار بخش اصلی تشکیل شده است که عبارت اند از:

۱. زبان مشترک زمان اجرا (Common Language Runtime): CLR به عنوان هسته مرکزی چارچوب دات نت، مدیریت اجرای برنامه ها را برعهده دارد و در واقع CLR همان نقش ماشین مجازی جاوا (Java Virtual Machine) را در سیستم های مبتنی بر جاوا بازی می کند. CLR علاوه بر اجرای کد، وظایف دیگری از جمله مدیریت حافظه و مدیریت نخ (Thread) را نیز انجام می دهد.



۲. کتابخانه کلاس چارچوب (Framework Class Library): این بخش حاوی مجموعه‌ای عظیم از کلاس‌ها است که برنامه‌نویسان می‌توانند از آن‌ها استفاده کنند و با سرعت بیش‌تری برنامه‌های خویش را توسعه دهند.
  ۳. سیستم نوع مشترک (Common Type System): این بخش باعث ایجاد هماهنگی بین زبان‌های مختلف چارچوب دات‌نت می‌شود.
  ۴. مشخصات زبان مشترک (Common Language Specification): CLS مجموعه‌ای از ویژگی‌های عمومی را که بین زبان‌های مختلف چارچوب دات‌نت مشترک است، توصیف می‌کند. چارچوب دات‌نت، جهت پشتیبانی از ویژگی چند زبانی از CTS و CLS استفاده می‌کند.
- CLS کد تمام زبان‌های مختلف چارچوب دات‌نت را به **زبان سطح میانی مایکروسافت** (Microsoft Intermediate Language) و یا به اختصار MSIL تبدیل می‌کند. به عنوان مثال، می‌توان یک کلاس را توسط زبان Visual Basic ایجاد کرد و در زبان C# از آن ارث‌بری کرد. در ادامه، با مفهوم کد MSIL بیش‌تر آشنا خواهید شد.

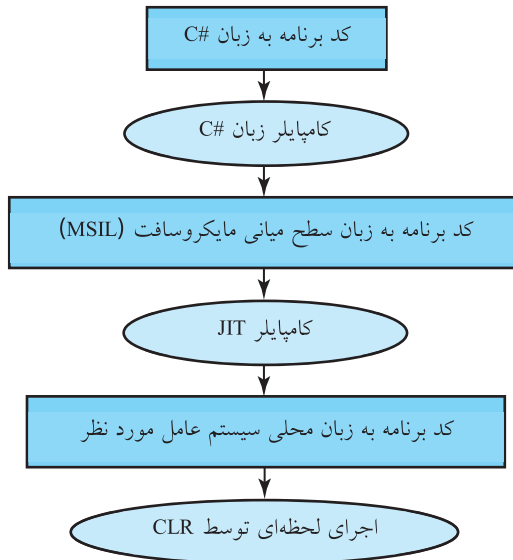
## ترجمه برنامه و کد MSIL

یک برنامه که به زبان C# نوشته شده است قبل از اجرا باید به کدی تبدیل شود که برای سیستم عامل قابل فهم باشد. به این کد، **کد محلی** (Native Code) می‌گویند. تبدیل یک کد از هر زبانی به کد محلی را کامپایل کردن می‌گویند و عملی است که توسط کامپایلر انجام می‌شود. در چارچوب دات‌نت عمل کامپایل از دو مرحله تشکیل شده است:

۱. **تبدیل کد برنامه به کد MSIL:** هنگامی که برنامه‌ای را کامپایل می‌کنید، بلافاصله کد محلی تولید نمی‌شود. در عوض کد شما به زبانی به نام MSIL تبدیل می‌شود. این کد برای سیستم عامل خاصی نیست و همچنین منحصر به زبان C# نیز نیست. به عبارت دیگر، کد زبان‌های دیگر چارچوب دات‌نت نیز هنگام کامپایل به MSIL تبدیل می‌شوند. MSIL یک زبان شیء‌گرا با گرامری مشابه زبان Assembly است.
  ۲. **تبدیل کد MSIL یک برنامه به کد محلی سیستم عامل:** این مرحله، وظیفه کامپایلر **Just In Time** و یا به اختصار کامپایلر **JIT** است. این کامپایلر کد MSIL یک برنامه را دریافت کرده و آن را به کدی تبدیل می‌کند که به وسیله سیستم عامل قابل اجرا باشد. بعد از اینکه این تبدیل توسط کامپایلر JIT انجام شد، سیستم عامل می‌تواند برنامه را اجرا کند. همان‌طور که از نام این کامپایلر نیز مشخص است کدهای MSIL فقط هنگامی به زبان محلی قابل فهم برای سیستم عامل تبدیل می‌شوند که بخواهند اجرا شوند.
- در گذشته برای اجرای یک برنامه بر روی سیستم عامل‌های مختلف نیاز بود که برای هر سیستم عامل، کد برنامه را یک مرتبه به صورت کامل کامپایل کنید. اما در چارچوب دات‌نت نیازی به این کار نیست. زیرا، برای هر نوع پردازنده و نیز هر نوع سیستم عامل یک نسخه از کامپایلر JIT وجود دارد. برنامه شما در هر سیستم عاملی که اجرا شود، کامپایلر JIT موجود در آن سیستم عامل کد MSIL برنامه شما را (که مستقل از سیستم عامل و نوع پردازنده است) دریافت کرده و کد محلی مناسبی تولید می‌کند تا برای سیستم عامل قابل فهم باشد.
- شکل ۱-۳، مراحل کامپایل یک برنامه نوشته شده با زبان C# را نشان می‌دهد. همان‌طور که در شکل ۱-۳ نشان داده شده است، کد برنامه شما هنگام کامپایل ابتدا به کد MSIL تبدیل می‌شود، سپس این کد قبل از اجرا توسط کامپایلر JIT به کد محلی تبدیل شده و کد محلی به وسیله CLR اجرا می‌شود (با فرض اینکه CLR که مهم‌ترین بخش چارچوب دات‌نت است، قبلاً در آن کامپیوتر نصب شده باشد).

## معرفی زبان C#

شرکت **سان مایکروسیستمز** (Sun Microsystems) در سال ۱۹۹۹ طی قراردادی اجازه استفاده از زبان Java را



شکل ۱-۳ مراحل کامپایل یک برنامه در چارچوب دات‌نت

در اختیار شرکت میکروسافت قرار داد تا در سیستم عامل خود از آن استفاده کند. همان‌طور که می‌دانید زبان Java در اصل به هیچ پلتفرم (Platform) یا سیستم عامل خاصی وابسته نبود ولی شرکت میکروسافت برخی از مفاد قرارداد را زیر پا گذاشت و قابلیت مستقل از سیستم عامل بودن را از زبان Java حذف کرد. شرکت سان میکروسیستمز نیز پرونده‌ای علیه شرکت میکروسافت تشکیل داد و در نهایت شرکت میکروسافت مجبور شد تا زبان برنامه‌نویسی جدیدی ایجاد کند. در ابتدا نام این زبان، COOL (C like Object Oriented Language) بود، هر چند در جولای سال ۲۰۰۰، زمانی که شرکت میکروسافت این پروژه را عمومی کرد، نام آن را به C# تغییر داد.

زبان C# به صورت اختصاصی برای کار با چارچوب دات‌نت طراحی شده است. آخرین نسخه این زبان، نسخه ۲۰۱۳ است که همزمان با چارچوب دات‌نت ۴/۵/۱ در اکتبر سال ۲۰۱۳ منتشر شد.

**آندرس هجلزبرگ** (Anders Hejlsberg) طراح زبان برنامه‌نویسی Del-phi، سرپرستی تیم طراحان زبان C# را برعهده داشت. این زبان به شدت از زبان‌های C++، Java و Delphi تاثیر پذیرفته است. طراحان زبان C# با الگوبرداری از مزایای زبان‌های C++ و Java و نادیده گرفتن برخی از امکانات تامل‌برانگیز و کم استفاده در هر یک از زبان‌های فوق، یک زبان برنامه‌نویسی مدرن و شیء‌گرا طراحی کرده‌اند. در مواردی حذف یا تغییر برخی از ویژگی‌های کم استفاده و درست درک نشده در هر یک از زبان‌های فوق، زمینه‌ساز ایجاد یک زبان ایمن و آسان شده است. به عنوان مثال، زبان C++ می‌تواند مستقیماً با استفاده از اشاره‌گرها عملیات دلخواه خود را در حافظه انجام دهد، اما در صورتی که عملیاتی این‌چنین به درستی کنترل و هدایت نگردند، خود می‌تواند باعث بروز مشکلات فراوانی شود. طراحان زبان C#، با درک اهمیت این موضوع، این ویژگی



شکل ۱-۴ آندرس هجلزبرگ سرپرست تیم طراحان زبان C#

را کماکان در آن قرار داده‌اند ولی به منظور جلوگیری از استفاده نادرست و ایجاد اطمینان‌های لازم، مسأله حفاظت را نیز در این زبان مورد توجه قرار داده‌اند. جهت استفاده از ویژگی فوق، برنامه‌نویسان می‌بایست با صراحت و به روشنی درخواست خود را از طریق استفاده از واژه‌های کلیدی مربوطه اعلان کنند (فراخوانی یک توانایی و استفاده از آن).

یادگیری زبان C# برای افرادی که سابقه آشنایی با یکی از زبان‌های برنامه‌نویسی ++C و یا Java را دارند کار مشکلی نخواهد بود. حتی افرادی که آشنایی اولیه با زبان اسکریپتی JavaScript و یا دیگر زبان‌های برنامه‌نویسی نظیر Visual Basic را دارند، به راحتی می‌توانند به زبان C# مسلط شوند.

## دلایل انتخاب زبان C# برای یادگیری

زبان C# همزمان با طراحی چارچوب دات‌نت و با در نظر گرفتن مفاهیم و نیازهای جدید برنامه‌نویسی طراحی شده است و به همین جهت است که شرکت مایکروسافت زبان C# را مادر چارچوب دات‌نت معرفی کرده است. این مورد به تنهایی می‌تواند دلیل قانع‌کننده‌ای برای یادگیری این زبان باشد، اما دلایل متعدد دیگری نیز وجود دارد که در ادامه به برخی از آن‌ها اشاره می‌شود:

- به عنوان یک زبان استاندارد برنامه‌نویسی توسط **انجمن تولیدکنندگان کامپیوتر اروپا (ECMA)** با شماره استاندارد ECMA-334 در سال ۲۰۰۱ ثبت شده است.
- به عنوان یک زبان استاندارد برنامه‌نویسی توسط **سازمان بین‌المللی استانداردها (ISO)** با شماره استاندارد ISO/IEC 23270 در سال ۲۰۰۳ ثبت شده است.
- در حال استاندارد شدن توسط موسسه ANSI است.
- طبق فهرست منتشر شده در **سایت معتبر تیوب (www.tiobe.com)**، زبان C# یکی از محبوب‌ترین زبان‌های برنامه‌نویسی است. این فهرست هر ماه یک بار بر مبنای کار مهندسان ماهر، دوره‌های آموزشی برگزار شده و فروش زبان‌ها بروز می‌شود و حتی از موتورهای جستجو نظیر گوگل، یاهو و آمازون نیز برای محاسبه رتبه‌بندی استفاده می‌شود.
- زبان C# از زبان‌های ++C و Java متولد شده است. بنابراین، برنامه‌نویسانی که قبلاً با این زبان‌ها آشنایی داشته‌اند به راحتی می‌توانند به زبان C# مسلط شوند.
- آینده صنعت نرم‌افزار در دستان چارچوب دات‌نت است و چون زبان C# به عنوان مادر چارچوب دات‌نت شناخته می‌شود، از اهمیت بسزایی برخوردار است و از پشتیبانی شرکت مایکروسافت برخوردار خواهد بود.
- منابع مناسب (فارسی و زبان اصلی) برای یادگیری زبان C# فراوان است.
- کدها و مثال‌های کاربردی برای زبان C# فراوان است.
- برای پیاده‌سازی یک قطعه کد در زبان C# به حجم کدنویسی کم‌تری نیاز است. به عنوان مثال، خیلی از جملات در زبان ژاپنی ممکن است تنها به دو کلمه احتیاج داشته باشند در حالی که در برخی زبان‌ها مثل زبان فارسی شاید نتوان اغلب این جملات را با تنها دو کلمه بیان کرد.
- **توسعه سریع نرم‌افزار (Rapid Application Development)** در زبان C# امکان‌پذیر است.

**خودآزمایی**

۱. چند مورد از مزایا و معایب سبک برنامه‌نویسی ساخت‌یافته را بیان کنید.
۲. چند مورد از تفاوت‌های سبک‌های برنامه‌نویسی ساخت‌یافته و شیء‌گرا را بیان کنید.
۳. سه اصل نهان‌سازی، وراثت و چندریختی را همراه با مثال شرح دهید.
۴. زبان‌های برنامه‌نویسی به چند دسته تقسیم می‌شوند؟
۵. چند مورد از ویژگی‌های زبان برنامه‌نویسی C# را بیان کنید.
۶. اجزای اصلی چارچوب دات‌نت را نام برده و هرکدام را مختصراً شرح دهید.
۷. چهار زبان رسمی چارچوب دات‌نت را نام ببرید.
۸. مراحل کامپایل یک برنامه در چارچوب دات‌نت را شرح دهید.