

فصل

معرفی زبان C



❖ اهداف آموزشی

- ❖ آشنایی با ویژگی‌های زبان برنامه‌نویسی C
- ❖ بررسی معایب زبان برنامه‌نویسی C
- ❖ آشنایی با انواع خطاها در زبان برنامه‌نویسی C
- ❖ آشنایی با ساختار زبان برنامه‌نویسی C به همراه جزئیات آن

▣ زبان C

▣ تاریخچه

زبان C توسط دنیس ریچی در آزمایشگاه‌های شرکت بل طراحی شد و اولین بار در سال ۱۹۷۲ بر روی کامپیوتر پیاده‌سازی شد. C در آغاز به‌عنوان زبان طراحی سیستم عامل Unix طراحی شده بود اما امروزه بیشتر سیستم عامل‌ها به زبان C یا C++ نوشته می‌شوند (از جمله سیستم عامل Windows).

▣ ویژگی‌های زبان C

■ C یک زبان ساخت‌یافته است

یعنی در آن امکان استفاده از توابع، بلاک‌ها و حلقه‌ها وجود دارد و می‌توان برنامه را به بخش‌های مختلفی تقسیم کرد و سپس بین این بخش‌ها ارتباط برقرار نمود. این کار علاوه بر خوانایی و زیبایی برنامه باعث سهولت در آزمایش و اشکال‌زدایی برنامه‌ها می‌شود. البته باید توجه کنید در زبان‌های ساخت‌یافته^۱ حتی‌الامکان از دستورات پرش مانند goto استفاده نشود.

۱- مجموعه دستورات روبه‌رو در نظر گرفته و گزینه مناسب را انتخاب نمایید.

for ...	(۱) احتمال آن که این زبان ساخت‌یافته باشد خیلی کم است.
for ...	(۲) زبان فوق ساخت‌یافته (structural) است ولی طرز برنامه‌نویسی، به
begin	شکل ساخت‌یافته نبوده است.
:	
if(problem)	(۳) زبان فوق نمی‌تواند ساخت‌یافته باشد.
goto error	
else	(۴) نمی‌توان در این مورد تصمیم‌گیری نمود.
:	
endif	
end	
:	
error :	
:	

1. Structural



■ C یک زبان پیمانه‌ای (ماژولار) است

بدین معنی که می‌توان برنامه را به بلوک‌ها (توابع) مجزا تقسیم کرد که هر بلوک از بلوک دیگر جدا بوده و اطلاعات آنان از یکدیگر مخفی می‌باشند. این خاصیت باعث ساده‌تر شدن برنامه‌ها و همچنین عیب‌یابی راحت‌تر برنامه‌ها می‌شود.

■ C یک زبان سطح میانی است

زبان اسمبلی یک زبان سطح پایین است یعنی به زبان ماشین (0, 1) و سخت‌افزار بسیار نزدیک است. زبان C که فاصله بیشتری از زبان ماشین دارد زبان **سطح میانی**^۱ نامیده می‌شود یعنی هم ویژگی‌های زبان‌های سطح پایین را دارد و هم از برخی امکانات زبان‌های سطح بالا برخوردار است. زبان‌های سطح بالا همچون پاسکال و بیسیک قواعد آسان‌تری دارند ولی در عوض قدرت و انعطاف‌پذیری آنها کمتر است.

■ قابلیت انعطاف

C یک زبان قابل انعطاف^۲ می‌باشد یعنی می‌توان بعضی عملیات کنترلی حساسی را که معمولاً با زبان اسمبلی قابل انجام است را توسط C انجام داد همچنین امکانات بسیار گسترده و پیچیده در سیستم عامل‌ها را می‌توان توسط آن پیاده‌سازی نمود.

■ C یک زبان همه‌منظوره^۳ است

زبان C در امور تجاری، مهندسی، حسابداری، بانک اطلاعاتی، سیستمی، گرافیک و بسیاری از امکانات کاربردی دیگر قابل استفاده است و به‌این خاطر C را یک زبان **همه‌منظوره** می‌نامند.

■ قابلیت حمل^۴

بعضی برنامه‌های نوشته شده بر روی یک سیستم را می‌توان بدون تغییر و یا با تغییراتی اندک روی سیستم‌های دیگر اجرا نمود.

۲- کدام گزینه در مورد ویژگی‌های زبان C درست است؟ زبان C:

- ۱) قابل حمل (portable) می‌باشد و یک زبان سطح میانی (middle-level) شناخته می‌شود.
- ۲) قابل حمل (portable) می‌باشد و یک زبان سطح بالا (high-level) شناخته می‌شود.

1. Medium Level
2. Flexible
3. All Purpose
4. Portability



۳) قابل حمل (portable) نمی‌باشد و یک زبان سطح میانی (middle-level) شناخته می‌شود.

۴) قابل حمل (portable) نمی‌باشد و یک زبان سطح بالا (high-level) شناخته می‌شود.

■ حساس به متن^۱

در زبان C حروف کوچک و بزرگ با هم متفاوت است، مثلاً سه متغیر CPP، cPP و cpp با هم یکسان نیستند. معمولاً بهتر است که تمام متغیرهای برنامه با حروف کوچک تایپ شوند.

■ تعداد کلمات کلیدی^۲ C کم است

این که زیاده‌تر بودن کلمات کلیدی یک زبان آن را قدرتمندتر خواهد ساخت تصویری غلط است چون هرچه تعداد کلمات کلیدی بیشتر باشد کاربر بیشتر باید مراقب باشد تا آنها را به‌جای متغیر به کار نبرد. مثلاً زبان بیسیک دارای ۱۵۰ کلمه کلیدی است ولی زبان C با داشتن تنها حدود ۳۲ کلمه کلیدی بسیار قدرتمندتر از آن است.

■ کد کوچک و سریع

زبان C معمولاً نسبت به زبان‌های دیگر کد کوچک‌تر و سریع‌تری تولید می‌کند.

■ توابع کتابخانه‌ای قوی

C دارای یک کتابخانه قوی شامل توابع مفید است این توابع در واقع قطعه برنامه‌هایی هستند که سازندگان C آنها را نوشته‌اند تا در صورت نیاز از آنها استفاده شود و نیاز به دوباره نوشتن آنها نباشد.

■ C یک زبان کامپایلری است

در زبان‌های مفسری مانند بیسیک هر خط از برنامه ابتدا بررسی شده و اگر خطای گرامری نداشته باشد به سراغ خط بعد می‌رود. در مقابل زبان‌های کامپایلری کل برنامه را یکجا بررسی می‌کنند و در صورت عدم وجود خطا آن را به کد ماشین تبدیل می‌کنند. زبان C یک زبان کامپایلری است ابتدا برنامه را تایپ می‌کنید و سپس با فشردن دکمه کامپایل تمام خطاهای برنامه به شما اطلاع داده خواهد شد.

■ دسترسی به سخت‌افزار

C می‌تواند به سخت‌افزار دسترسی یابد و حتی بر یک لیست از حافظه نیز کنترل داشته باشد برای این کار نیاز به کدنویسی طولانی نیست زیرا C دارای عملگرهای متعددی می‌باشد.

1. Case Sensitive

2. Keywords



۳- کدام گزینه صحیح است؟

- ۱) زبان برنامه‌نویسی C مانند زبان پاسکال حساس به حروف (Case Sensitive) است.
- ۲) قابل حمل بودن، یعنی پس از کامپایل کردن یک برنامه می‌توان آن برنامه را بدون تغییر روی هر کامپیوتر و سیستم‌عامل دیگر اجرا نمود.
- ۳) تعداد کلمات کلیدی (Keywords) ارتباطی به قدرت یک زبان برنامه‌نویسی ندارد.
- ۴) در زبان برنامه‌نویسی C امکان دسترسی و بهره‌برداری از سخت‌افزار وجود ندارد.

□ ویژگی‌های مخاطره‌آمیز زبان C

- ۱- استفاده گسترده از زبان C روی کامپیوترهای مختلف متأسفانه منجر به طراحی و تولید نسخه‌های متعددی از آن شد مانند: C سنتی یا کلاسیک، Turbo C، Borland C و ... به‌همین دلیل مؤسسه استاندارد ملی آمریکا (ANSI) با همکاری سازمان استاندارد جهانی (ISO) C را به‌صورت جهانی استاندارد ساخت و آن را تحت عنوان ANCI C ارائه نمود.
- ۲- عملیات تبدیل نوع داده‌ها در زبان C بسیار راحت است اما همین مزیت می‌تواند به‌راحتی تبدیل به نقطه ضعف آن شود زیرا ممکن است یک سری تبدیل داده انجام گیرد که شما انتظار آن را ندارید و این مسأله موجب دریافت نتیجه‌هایی دور از انتظار ما شود. (خطاهای منطقی)
- ۳- مختصر شدن زبان C همراه با در اختیار داشتن عملگرهای فراوان فهم برنامه را مشکل کرده است. بعلاوه توانایی C در استفاده از اشاره‌گرها شما را در مسیری قرار می‌دهد که در آن امکان اشتباه برنامه‌نویسی افزایش می‌یابد و اشکال‌زدایی مشکل‌تر می‌شود.

□ انواع خطا در زبان C

■ خطاهای نحوی^۱

به خطاهای گرامری دستورات (غلط تایپی) خطای نحوی گفته می‌شود. این خطاها در زمان ترجمه برنامه آشکار می‌شوند و به‌همین خاطر به آنها Compile Time Errors نیز گفته می‌شود. اگر برنامه‌ای دارای خطاهای نحوی باشد به کد ماشین تبدیل نخواهد شد. مثلاً اگر برنامه‌نویس به‌جای دستور while یکی از عبارتهای Waile یا WHILE را بنویسد یا علامت ; را انتهای دستورات C نگذارد آنگاه مرتکب یک خطای نحوی شده است.

1. Syntax Errors



۴- قرار ندادن نقطه ویرگول (semicolon) در انتهای یک دستور به منزله وجود کدام یک از خطاهای زیر است؟

(۱) خطای زمان اجرا (run-time error)

(۲) خطای نحوی (syntax error)

(۳) خطای منطقی (logical error)

(۴) به هر دو دسته خطای نحوی و خطای منطقی تعلق دارد.

■ خطاهای زمان اجرا^۱

پس از آن که برنامه کامپایل شده اجرا شد ممکن است در حین اجرا دچار خطایی همچون خطای تقسیم بر صفر و یا جذر اعداد منفی شود که این قبیل خطاها را خطای زمان اجرا می‌گویند. مثلاً اگر در هنگام اجرای دستور $x=a/b$; مقدار متغیر b برابر با صفر باشد یک خطای تقسیم بر صفر با پیغام خطای Division By Zero صادر می‌شود.

■ خطاهای منطقی^۲

پس از آن که برنامه کامپایل شد و اجرا گردید ممکن است دچار خطای اجرا نشود ولی آن جوابی که مورد نظر شما است و از آن انتظار دارید را تولید نکند به این قبیل خطاها، خطاهای مفهومی یا معنایی نیز می‌گویند. رفع این قبیل خطاها بسیار مشکل است. مثلاً فرض کنید که برنامه‌نویس به جای دستور $x=x/10$; اشتباهاً دستور $x=x\%10$ تایپ کرده باشد. بدیهی است که هیچ خطای نحوی مرتکب نشده است ولی جواب نهایی چیز دیگری خواهد بود. نوع دیگر ایجاد این گونه خطاها مغایرت تفکر برنامه‌نویس با منطق کامپایلر است مثلاً اگر برنامه‌نویس اقدام به چاپ یک متغیر محلی نماید کامپایلر بدون توجه به نوع آن متغیر آدرس از حافظه را که به آن اختصاص داده را چاپ می‌نماید و یا اگر برنامه‌نویس عدد 12.8 را به یک متغیر صحیح نسبت دهد کامپایلر فقط عدد 12 را به آن متغیر نسبت می‌دهد.

□ ساختار برنامه در C

برای آشنایی شما با اجزای اساسی یک برنامه به زبان C مثالی ساده‌ای را در این بخش مطرح می‌کنیم. برنامه زیر عددی صحیح به نام n را از صفحه کلید دریافت کرده و سپس حاصل تابع $y=3n-5$ را محاسبه کرده و نهایتاً عدد حاصله را روی مانیتور نشان می‌دهد.

1. Run Time Errors

2. Logical Errors



```

1) #include <stdio.h>
2) /*This Program calculates the result of
3) y=3n-5 and Prints the result on minitor*/
4) void main( )
5) {
6) int x,y;
7) printf("Please enter x:");
8) scanf("%d", &x);
9) y=3*x-5;
10) printf("\n y=%d", y);
11) }

```

حال با توجه به شماره خطوط جزئیات برنامه را شرح می‌دهیم:

■ شرح خط ۱

دستور `#include <stdio.h>` اطلاعات موجود در فایل کتابخانه `stdio.h` را در اختیار برنامه شما قرار می‌دهد. در این مازول اطلاعات ورودی و خروجی استاندارد (یعنی صفحه کلید و مانیتور) قرار دارد. `stdio` مخفف کلمات `standard input/output` می‌باشد. دستوراتی همچون `include` که با علامت `#` آغاز می‌شوند دستورات پیش‌پردازنده^۱ نام دارند. این دستورات مربوط به کامپایلر بوده و تبدیل به زبان ماشین (1,0) نمی‌شوند. برای افزودن چند سر تیر (header) به برنامه باید از `#include` استفاده کنید دستور `include` به دو صورت زیر قابل استفاده است:

`#include <نام هدر فایل>`

و

`#include "نام هدر فایل"`

دقت کنید در انتهای دستورات پیش‌پردازنده نباید سمی کالن (;) قرار دهید. اگر نام فایل بین دو علامت " " قرار بگیرد هنگام جستجو ابتدا فهرست جاری و سپس فهرست‌های تعیین شده در منوی `options→directories` بررسی می‌شود. اگر نام فایل بین علامت < > باشد ابتدا مسیرهای تعیین شده در منوی `options→directories` جستجو می‌شوند و سپس اگر فایل پیدا نشد آن‌گاه فهرست جاری جستجو می‌شود.

توجه: بهتر است که کاربر آن دسته از فایل‌های `header` که خودش نوشته است را داخل علامت " " قرار دهد و فایل‌هایی متعلق به خود سیستم را در علامت < > قرار دهد.



■ شرح خطوط ۲ و ۳

در ANCI C اگر بخواهید توضیحات^۱ به برنامه خود اضافه کنید می‌توانید آن را بین دو علامت `/*` /`*/` قرار دهید. با این روش می‌توانید چند خط توضیحات اضافه کنید. در Turbo C به غیر از علامت فوق می‌توانید از نماد `//` نیز برای ارائه توضیحات استفاده کنید. تنها تفاوت در این است که با نماد `//` فقط می‌توانید در یک خط توضیحات را ارائه کنید ولی با نماد `/*` /`*/` می‌توانید چندین خط توضیحات را ارائه کنید.

۵- کدام عبارت در مورد علایم درج Comment صحیح‌تر است؟

- (۱) علامت `//` برای مواردی که Comment از یک خط بیشتر نیست.
- (۲) علامت `//` برای مواردی که Comment بیشتر از یک خط است.
- (۳) علامت `/*` /`*/` برای مواردی که Comment از یک خط بیشتر است.
- (۴) علامت `{ }` برای همه موارد

■ شرح خط ۴

هر برنامه در زبان C در واقع یک تابع است که با نام `( main )` شروع می‌شود و کلمه `void` که قبل از آن نوشته شده است نیز بیان می‌کند که این تابع (برنامه) مقداری را بر نمی‌گرداند. البته اگر کلمه `void` نوشته نشود به‌طور پیش‌فرض `int` در نظر گرفته می‌شود و بایستی مقدار صحیحی برگردانده شود، به‌همین دلیل در انتهای برنامه حتماً باید دستور `return 0;` نوشته شود. دقت کنید این مقدار به سیستم‌عامل برگردانده می‌شود. خود تابع `( main )` می‌تواند شامل چندین زیر تابع باشد که این ویژگی ایده برنامه‌نویسی ماژولار را شکل می‌دهد.

۶- در زبان برنامه‌نویسی C شروع برنامه با کدام تابع زیر آغاز می‌شود؟

- (۱) `int` (۲) `#printf` (۳) `main` (۴) `{ }`

■ شرح خطوط ۵ و ۱۱

آکولادهای باز و بسته به ترتیب ابتدا و انتهای بدنه تابع را مشخص می‌کند در هر برنامه به تمام دستوراتی که بین `{ }` قرار می‌گیرند **بلوک**^۲ گفته می‌شود. دستوراتی همچون `for`، `if`، `while` و ... دستوراتی بلوکی هستند. بیشتر دستورات در زبان C به علامت `;` ختم می‌شوند.

1. Comments

2. Block



■ شرح خط ۶

عبارت `int x,y;` دو متغیر به نام‌های `x` و `y` را از نوع عدد صحیح تعریف می‌کند این دستور می‌توانست به صورت ۲ دستور جداگانه نیز نوشته شود یعنی `int x` و `int y`.

■ شرح خط ۷

دستور `printf` مخفف کلمه `print format` می‌باشد و برای خروجی استاندارد روی مانیتور به کار می‌رود. در این دستور هر آنچه درون گیومه " " قرار بگیرد عیناً چاپ می‌شود (به استثنای شاخص‌های تبدیل که همراه با علامت % می‌آیند). الگوی `printf` در فایل `stdio.h` قرار دارد که در خط ۱ توضیح داده شد.

■ شرح خط ۸

دستور `scanf` مخفف کلمه `scan format` می‌باشد و در فایل `stdio.h` قرار دارد. وظیفه این دستور خواندن اطلاعات از ورودی استاندارد (صفحه کلید) است. در این دستور علامت & باید قبل از نام متغیر قرار گیرد. %d شاخص تبدیل نام دارد و بدین معنی است که عدد ورودی باید عددی صحیح باشد (راجع به شاخص‌های تبدیل در فصول بعدی توضیح بیشتری خواهیم داد).

■ شرح خط ۹

طبق این دستور ابتدا 3 در `x` ضرب شده سپس منهای 5 می‌شود و مقدار نهایی داخل `y` قرار می‌گیرد.

■ شرح خط ۱۰

کد `\n` در دستور `printf` باعث رفتن مکان‌نما به ابتدای خط بعد روی صفحه مانیتور می‌شود، سپس در آن‌جا عبارت `y=` عیناً چاپ شده و بعد عددی صحیح در سمت راست آن ظاهر می‌شود مثلاً اگر مقدار `x` را برابر 1 قرار دهیم خروجی زیر را بر روی صفحه مانیتور مشاهده خواهید نمود :

`y = -2`

نکته: برای دستور انتساب از علامت `=` و برای انجام مقایسه از علامت `==` استفاده می‌شود.

(عملگرهای مقایسه‌ای در فصل ۳ توضیح داده می‌شود)

نکته: در زبان C هر دستور حداکثر می‌تواند تا 255 کاراکتر را در خود جای دهد.

نکته: اگر چندین تابع در برنامه داشته باشیم برنامه همیشه ابتدا از تابع `main` شروع می‌شود بنابراین نوشتن این تابع الزامی و بقیه توابع اختیاری است.

۷- در زبان C برنامه الزاماً با تابع شروع می‌شود؟

(۱) `scanf` (۲) `printf` (۳) `main` (۴) `getche`

ک پاسخنامه فصل اول

۱- گزینه (۲) این برنامه چون از بلوک‌های جداگانه (تودرتو) تشکیل شده، ساخت‌یافته می‌باشد ولی چون در آن از دستور پرش (goto) استفاده شده، طرز برنامه‌نویسی ساخت‌یافته نمی‌باشد.

۲- گزینه (۱) C یک زبان قابل حمل (Portability) می‌باشد یعنی برنامه‌ای نوشته شده بر روی آن را می‌توان بدون تغییر یا با تغییراتی اندک روی سیستم‌های دیگر اجرا نمود، همچنین C یک زبان سطح میانی است یعنی هم خصوصیات زبان‌های سطح بالا را دارد و هم خصوصیات زبان‌های سطح پایین (زبان ماشین).

۳- گزینه (۳) زبان پاسکال Case sensitive نیست. قابل حمل بودن یک زبان بدین معنی است که خود برنامه (Source) بر روی کامپیوترها و سیستم عامل‌های دیگر اجرا شود نه برنامه کامپایل شده و در نهایت زبان C دارای عملگرهایی است که امکان دسترسی به سخت‌افزار را امکان‌پذیر می‌کند.

۴- گزینه (۲) اگر انتهای دستورات C سمی‌کالن (;) قرار ندهیم، در زمان کامپایل پیام خطا صادر می‌شود. (خطای نحوی)

۵- گزینه (۱) با استفاده از علامت // تنها یک خط توضیحات (Comments) و با استفاده از علامت /**/ یک یا چند خط توضیحات را نوشت. از علامت { } به هیچ عنوان نمی‌توان برای توضیحات استفاده کرد.

۶- گزینه (۳)

۷- گزینه (۳) اگر چندین تابع در یک برنامه داشته باشیم باز هم برنامه از تابع main اجرا می‌شود، حتی اگر قبل از main تابع دیگری وجود داشته باشد.



۸ خودآزمایی فصل اول

- ۱- هفت مورد از ویژگی‌های زبان C را نام برده و سه مورد از آنها را به اختصار توضیح دهید.
- ۲- ویژگی‌های مخاطره‌آمیز زبان C چیست؟ توضیح دهید.
- ۳- انواع خطاها در زبان C را نام برده و هر یک را به اختصار توضیح دهید.
- ۴- کدام یک از عبارات زیر صحیح است؟ دلیل آن را توضیح دهید.
 - الف) علامت پیش‌پردازنده در زبان C علامت # است.
 - ب) در زبان C فایل‌های برنامه‌ای دارای پسوند (C). قابل حمل است.
 - ج) در زبان C برنامه همواره از تابعی شروع می‌شود که نام آن main است.
 - د) دستور کامل معرفی هدر فایل در زبان C به صورت `include stdio.h` می‌باشد.
- ۵- تابع اصلی () `main` در زبان C به چه معناست؟
- ۶- کدام یک از موارد زیر صحیح است؟ دلیل آن را توضیح دهید.
 - الف) زبان C یک زبان مفسری است.
 - ب) در زبان C انتهای دستورات پیش‌پردازنده سمی کالن (:) قرار نمی‌گیرد.
 - ج) در زبان C اگر توضیحات (Comments) بیشتر از یک خط باشد از نماد // استفاده می‌شود.
 - د) در زبان C هر دستور می‌تواند بیش از ۲۵۵ کاراکتر داشته باشد.



انواع داده‌ها، متغیرها و ثابت‌ها

❖ اهداف آموزشی

- ❖ انواع داده‌ها در زبان C
- ❖ آشنایی با پیشندهای تغییردهنده نوع و شرح آنها
- ❖ بررسی نحوه ورودی و خروجی در زبان C
- ❖ آشنایی با قوانین نامگذاری متغیرها و مقداردهی آنها
- ❖ انواع مقادیر ثابت در زبان C
- ❖ آشنایی با نحوه تعریف ثابت و کاربرد آن در زبان C
- ❖ کاربرد ماکرو و تفاوت آن با تابع

□ انواع داده^۱

داده‌های اصلی در زبان C عبارتند از char, int, float, double و void. در جدول ۱-۲ مقدار فضای اشغال شده و محدوده آنها را مشاهده می‌کنید:

نوع داده	نوع داده ذخیره شده	مقدار فضای اشغال شده بر حسب بایت	محدوده قابل نمایش
char	کاراکتر یا عدد صحیح	1	-128 تا +127
int	عدد صحیح	2	-32768 تا +32767
float	عدد اعشاری (با ۷ رقم اعشار)	4	3.4E-38 تا 3.4E+38
double	عدد اعشاری مضاعف (با ۱۶ رقم اعشار)	8	1.7E-308 تا 1.7E38
void	_____	_____	_____

جدول ۱-۲

۱- انواع داده‌ها در زبان C استاندارد کدامند؟

۱) float, int, char, void

۲) void, char, int, double, float

۳) float, int, char

۴) float, int, char, unsigned int, unsigned char, void

نکته: در زبان C داده منطقی (Boolean) وجود ندارد و به‌جای آن از سایر انواع داده استفاده می‌شود یعنی اگر متغیر دارای مقدار صفر باشد معادل False و اگر غیر صفر باشد معادل True در نظر گرفته می‌شود. (در C++ نوع داده bool وجود دارد که مقادیر آن True و False می‌باشد و ۲ بایت حافظه اشغال می‌کند)



نکته: نوع داده‌های char و int به ترتیب برای ذخیره کاراکتر و عدد صحیح به کار می‌روند ولی در صورتی که به متغیر از نوع char مقدار عددی نسبت دهیم و یا متغیری از نوع صحیح را برابر کاراکتر قرار دهیم آن‌گاه زبان C این داده‌ها را به صورت استاندارد به یکدیگر تبدیل می‌کند. (کداسکی کاراکتر a برابر 97 است)

مثال :

char ch = 97 ; معادل → char ch = 'a';

int i = 'a' ; معادل → int i = 97;

۲- خروجی قطعه برنامه زیر کدام است؟

```
int i='a';
printf("%d",i);
```

(۱) کاراکتر i را نمایش می‌دهد. (۲) کاراکتر a را نمایش می‌دهد.

(۳) برنامه دارای خطا می‌باشد. (۴) عدد 97 معادل کداسکی a را نمایش می‌دهد.

نکته: نوع داده void دارای ۳ کاربرد زیر است:

۱- برای معرفی تابعی که چیزی بر نمی‌گرداند.

۲- برای معرفی تابعی که هیچ پارامتر ورودی ندارد.

۳- برای تعریف اشاره‌گر عمومی

مثال: تابع مقابل هیچ مقدار ورودی ندارد و مقداری را هم بر نمی‌گرداند:

```
void sub(void)
```

□ پیشوندهای تغییردهنده نوع^۱

در زبان C، ۴ نوع پیشوند وجود دارد که اگر آن‌ها را قبل از انواع داده اصلی ذکر کنیم انواع داده‌های جدیدی به وجود می‌آیند که به این پیشوندها، پیشوندهای تغییردهنده نوع یا اصلاح‌گر گفته می‌شود، که عبارتند از:

signed, unsigned, short, long

پیشوندهای signed و unsigned به ترتیب برای اعداد علامت‌دار و اعداد بدون علامت و پیشوندهای short و long برای داده‌های کوتاه و طولانی استفاده می‌شوند. برخی از این پیشوندها بر میزان فضای اشغال شده توسط آن داده نیز تأثیر می‌گذارند و یا حتی محدوده آن‌ها را نیز تغییر می‌دهند. ولی دقت کنید در زبان C پیشوند short بر هیچ یک از انواع داده‌ها اثر نمی‌گذارد یعنی میزان فضای اشغال شده توسط آن نوع داده تغییر نمی‌کند.