

Time	Narration
00:01	به برنامه آموزشی Increment and Decrement Operators in C and C++ خوش آمدید.
00:08	در این برنامه
00:10	اپراتورهای افزایشی و کاهشی
00:12	'++' مثل $a++$ که اپراتور پسوند افزایشی می باشد.
00:18	و $++a$ که اپراتور پیشوند افزایشی می باشد.
00:22	'--' مثل $a--$ که اپراتور پسوند کاهشی می باشد.
00:27	و $--a$ که اپراتور پیشوند کاهشی می باشد.
00:31	و در مورد typecasting یاد می گیریم.
00:35	برای ضبط این برنامه من از Ubuntu Operating System نسخه 11/10 و کامپایلر gcc و ++g نسخه 4/6/1 در
00:40	ubuntu استفاده کرده ام.
00:48	عامل ++ مقدار فعلی را با یک مقدار افزایش می دهد.
00:54	و $a++$ و $++a$ مساوی $a=a+1$ می باشند.
01:00	عامل - مقدار فعلی را با یک مقدار کاهش می دهد.
01:06	و $a--$ و $--a$ مساوی $a=a-1$ می باشند.
01:13	حالا من استفاده از عامل های افزایشی و کاهشی را به کمک یک برنامه C نشان می دهم.
01:19	من از قبل برنامه را نوشته ام پس کدها را توضیح می دهم.
01:25	در اینجا ما کد برای عامل های افزایشی و کاهشی در C داریم.
01:30	من یک متغیر عدد صحیح a که مقدار 1 را دارد انتخاب کرده ام.
01:35	با این روش قادر به دیدن تغییرات در مقدار a می باشیم.
01:39	و این به ما ایده بهتری در مورد عمل کردن اپراتورها می دهد.
01:47	بینیم که عامل postfix افزایشی چگونه کار می کند.
01:51	خروجی این دستور printf در اینجا 1 میباشد.
01:55	مقدار تغییر نمی کند.
01:57	و به این دلیل است که عملیات (postfix) پسوندی بعد از ارزیابی عامل ها انجام می شود.
02:04	اگر عملیاتی روی ++a انجام شود , روی مقدار فعلی a انجام می شود.

02:10	بعد از آن مقدار a افزایش می یابد.
02:17	حالا اگر مقدار a را اینجا ببینیم با عدد 1 افزایش یافته است.
02:27	ما دوباره به a عدد 1 را می‌دهیم و تغییرات را می بینیم.
02:35	حالا به اپراتورهای پیشوند افزایشی (prefix increment operators) می رویم.
02:38	دستور printf عدد 2 را روی صفحه چاپ می کند.
02:42	چون که عمل prefix (پیشوند) قبل از اینکه عامل ارزیابی شود اتفاق می افتد.
02:49	بنابراین مقدار a ابتدا با عدد 1 افزایش می‌یابد و سپس چاپ می شود.
02:58	ما دوباره مقدار a را چاپ می‌کنیم تا ببینیم که تغییرات دیگری وجود ندارد.
03:03	حالا با اجرای این کد ، بررسی می کنیم.
03:07	ما این خط را کامنت می‌دهیم. /*,*/ را تایپ کنید.
03:19	Save را کلیک کنید.
03:22	من فایل را با نام incrdecr.c ذخیره کرده ام.
03:29	پنجره ترمینال را با فشار دادن همزمان کلیدهای Ctrl,Alt و T در صفحه‌کلید باز کنید.
03:35	برای کامپایل gcc space incrdecr dot c space minus o space incr را در ترمینال تایپ کنید. Enter را فشار دهید.
03:51	برای اجرای کد ./incr را تایپ کنید. Enter را فشار دهید.
03:59	خروجی روی صفحه نمایش داده می شود.
04:01	وقتی که ++a را پرینت کنید، این خروجی می باشد.
04:06	و این خروجی زمانی که ++a را پرینت کنید.
04:09	و نتیجه به همان صورت که توضیح داده شد می باشد.
04:13	به بقیه برنامه برمی گردیم.
04:16	من حالا پسوند و پیشوندهای (postfix and prefix) کاهش را توضیح می دهم.
04:21	کامنتهای چند خطی را از اینجا و اینجا حذف می کنیم.
04:29	دوباره مقدار 1 را به a می دهیم.
04:35	دستور printf خروجی 1 را همانطور که قبلاً گفتم می دهد.

04:40	مقدار a وقتی که a-- ارزیابی شد کاهش خواهد یافت چون این یک عبارت پسوندی (postfix) می باشد.
04:47	بیانیه بعدی مقدار a را 0 چاپ می کند.
04:51	مقدار a با عدد 1 کاهش یافته است.
04:54	ما حالا عامل پیشوند کاهشی (prefix decrement) را داریم.
04:58	خروجی دستور printf اینجا 0 می باشد.
05:00	چون این یک عامل پیشوندی می باشد.
05:05	عمل پیشوندی قبل از اینکه عامل ارزیابی شود اتفاق می افتد.
05:09	خروجی دستور printf اینجا 0 می باشد.
05:11	هیچ تغییر دیگری در مقدار a صورت نمی گیرد.
05:15	حالا return 0; را تایپ کنید و آکولاد آخر را ببندید.
05:21	Save را کلیک کنید.
05:24	به ترمینال باز گردید.
05:27	برای کامپایل gcc space incrdecr dot c space minus o space incr را تایپ کنید. Enter را فشار دهید.
05:42	برای اجرا ./incr. را تایپ کنید. Enter را فشار دهید.
05:52	این خروجی زمانی که a-- را پرینت کنید می باشد.
05:56	این خروجی زمانی که --a را پرینت کنید می باشد.
05:59	حالا می بینیم که عامل های افزایشی و کاهشی چگونه عمل می کنند.
06:05	اگر بخواهیم همین برنامه را در C++ بنویسیم
06:07	در کدهای c تغییراتی ایجاد می کنم.
06:10	حالا به editor برمی گردم.
06:13	و این فایل C++ با کدهای لازم می باشد.
06:16	توجه کنید که header با header در فایل C فرق دارد.
06:20	ما همچنین دستور using namespace را داریم.
06:24	همچنین در C++ دستور خروجی cout می باشد.
06:28	بغیر از این تفاوتها کدها در هر دو خیلی شبیه به هم می باشند.

06:33	فایل را ذخیره کنید. فایل با امتداد <code>cpp</code> . ذخیره شده است.
06:40	کد را کامپایل می کنیم.
06:42	ترمینال را باز کنید و <code>g++ space incrdecr dot cpp space minus o space incr</code> را تایپ کنید. <code>Enter</code> را فشار دهید.
07:00	برای اجرا <code>./incr</code> . را تایپ کنید. <code>Enter</code> را فشار دهید.
07:07	خروجی روی صفحه نمایش داده می شود.
07:10	میبینیم که خروجی مثل همان در برنامه <code>C</code> می باشد.
07:15	حالا مفهوم <code>typecasting</code> را داریم.
07:17	این به یک روش در <code>C</code> و <code>C++</code> اجرا می شود.
07:22	که برای ساختن متغیر از یک نوع که به صورت نوع دیگر رفتار می کند , استفاده می شود
07:27	<code>Typecasting</code> با قرار دادن نوع داده‌ای که می‌خواهید در بین پرانتز انجام می شود.
07:33	و این قالب جلوی متغیری که می‌خواهید قرار داده می شود.
07:38	و این <code>typecast</code> تنها برای یک عمل اعتبار دارد.
07:42	حالا <code>a</code> مثل متغیر <code>float</code> تنها برای یک عمل رفتار می کند.
07:47	واینجا یک مثال است که من از قبل ایجاد کرده ام.
07:50	حالا کد را توضیح می دهم.
07:54	در ابتدا متغیر های <code>a</code> و <code>b</code> را عدد صحیح و <code>c</code> را اعشاری (<code>float</code>) اعلام می کنیم.
08:00	به <code>a</code> مقدار 5 و به <code>b</code> مقدار 2 را می دهیم.
08:06	و عملیات را روی <code>a</code> و <code>b</code> انجام می دهیم.
08:10	و <code>a</code> را تقسیم بر <code>b</code> میکنیم. و نتیجه در <code>c</code> ذخیره می شود.
08:14	ما <code>%.2f</code> را برای دقت 2 مکان اعشاری استفاده می کنیم.
08:20	نتیجه نمایش داده شده 2.00 برخلاف نتیجه انتظاری 2.50 می باشد.
08:25	قسمت اعشاری ناقص شده است چون <code>a</code> و <code>b</code> عدد صحیح می باشند.
08:31	برای تقسیم واقعی یکی از عاملها باید به (<code>float</code>) اعشاری <code>typecast</code> شود.
08:35	ما باید <code>a</code> را به <code>float</code> تغییر دهیم <code>c</code> مقدار واقعی تقسیم را نگه می دارد.

08:41	حالا نتیجه درست تقسیم نمایش داده می‌شود و جواب 2.50 می باشد.
08:47	و 0; return را تایپ کنید و آکولاد را نیز ببندید.حصح
08:51	Save را کلیک کنید.فایل را با امتداد c. ذخیره کنید.
08:55	من فایل خود را با نام typecast.c ذخیره کرده ام.
08:59	ترمینال را باز کنید.
09:01	برای کامپایل gcc space typecast dot c space minus 0 space type را تایپ کنید. Enter را فشار دهید.
09:17	برای اجرا ./type. را تایپ کنید . Enter را فشار دهید.
09:25	خروجی روی صفحه نمایش داده می شود.
09:27	با دیدن هر دو مقدار تأثیر typecasting را می بینیم.
09:32	حالا برنامه را خلاصه میکنیم.
09:34	در این برنامه یاد گرفتیم که
09:36	عملگرهای افزایشی و کاهشی را چگونه استفاده کنیم.
09:40	در مورد postfix و prefix یاد گرفتیم.
09:44	همچنین typecasting و چگونگی استفاده از آن را یاد گرفتیم.
09:47	بعنوان ارائه:
09:49	یک برنامه برای حل معادله زیر بنویسید a تقسیم بر b به اضافه c تقسیم بر d
09:56	مقادیر a, b, c و d بعنوان ورودی از کاربر گرفته شوند.
10:01	و typecasting را برای انجام درست تقسیم استفاده کنید.
10:05	ویدیو را در لینک زیر مشاهده کنید.
10:08	این خلاصه پروژه spoken tutorial می باشد
10:10	اگر پهنای باند خوبی ندارید، ابتدا دانلود سپس مشاهده کنید
10:15	تیم پروژه spoken tutorial کارگاه آموزشی استفاده از spoken tutorial ارائه می‌دهد
10:17	
10:20	و به کسانی که آزمون آنلاین را قبول شوند گواهینامه میدهد

10:24	برای جزئیات بیشتر لطفاً به "contact@spoken-tutorial.org" ایمیل بفرستید
10:33	Spoken tutorial بخشی از پروژه Talk To a Teacher می باشد.
10:37	که تحت پشتیبانی National Mission on Education از طریق ICT, توسط MHRD دولت هند می باشد.
10:44	اطلاعات بیشتر spoken hyphen tutorial dot org slash NMEICT hyphen intro موجود می باشد
10:55	ترجمه و صداگذاری توسط شبنم اقبال خان... با تشکر از شما