

Loops

Time	Narration
00:01	به برنامه آموزشی C++ و C Loops خوش آمدید.
00:06	در این برنامه این موارد را یاد می‌گیریم:
00:09	For loop, while loop
00:12	و do...while loop و این را با کمک مثال انجام می‌دهیم.
00:17	و همچنین تعدادی از اشتباهات رایج و راه حل آن را بررسی می‌کنیم.
00:21	برای ضبط این برنامه من از Ubuntu Operating System نسخه 11/10 و کامپایلر gcc و ++g نسخه 4/6/1 در Ubuntu استفاده کرده‌ام.
00:24	
00:28	
00:34	با مقدمه‌ای در مورد حلقه‌ها (loops) شروع می‌کنیم.
00:38	حلقه‌ها (loops) برای اجرای گروهی از دستورها به طور مکرر استفاده می‌شوند.
00:44	بستگی به نوع هدف آن‌ها به سه نوع تقسیم می‌شوند.
00:48	While loop do..... while loop
00:51	و for loop. با while loop شروع می‌کنیم.
00:56	While loop وضعیت را در ابتدا بررسی می‌کند.
01:00	ساختار آن (وضعیت) while می‌باشد.
01:03	در بین گروه‌ها statement block
01:07	حالا به do...while loop می‌رویم.
01:09	و do..while loop (حداقل یک بار) at least once قبل از اینکه وضعیت معتبر باشد اجرا می‌شود.
01:15	و ساختار آن این‌گونه است:
01:17	و do (بین گروه‌ها) statement block
01:20	بعد از گروه (وضعیت) while
01:23	می‌بینید که وضعیت در آخر بررسی می‌شود.
01:27	حالا یک مثال در while loop و do... while loop می‌بینیم.
01:32	از قبل کدها را در editor تایپ کرده‌ام.

01:35	حالا آن را باز می‌کنم.
01:37	نام فایل ما <code>while.c</code> می‌باشد.
01:41	حالا جمع کردن 10 عدد اول با استفاده از حلقه <code>while</code> را یاد می‌گیریم.
01:47	حالا کد را توضیح می‌دهم.
01:49	این <code>header file</code> ما می‌باشد.
01:51	در عملگر <code>main()</code> ما دو <code>integer variables x 0</code> و <code>y</code> را اعلام و به آنها مقدار اولیه 0 را داده ایم.
01:59	و این <code>while loop</code> ما می‌باشد.
02:02	وضعیت حلقه <code>while</code> در اینجا <code>x is less than or equal to 10</code> می‌باشد.
02:06	در اینجا مقدار <code>x</code> و <code>y</code> با هم جمع می‌شوند.
02:10	مقداری که بعد از جمع به دست می‌آید در <code>y</code> ذخیره می‌شود.
02:15	سپس مقدار <code>y</code> را چاپ می‌کنیم.
02:18	در اینجا <code>x</code> افزایش یافته است.
02:20	یعنی متغیر <code>x</code> با عدد یک افزایش یافته است.
02:25	و این دستور <code>return</code> می‌باشد.
02:27	حالا برنامه را اجرا می‌کنیم.
02:30	پنجره ترمینال را با فشار دادن همزمان کلیدهای <code>Ctrl,Alt</code> و <code>T</code> در صفحه‌کلید باز کنید.
02:39	و <code>gcc space while dot c space hyphen o space while</code> را تایپ کنید.
02:45	<code>Enter</code> را فشار دهید.
02:47	و <code>./while</code> را تایپ کنید. <code>Enter</code> را فشار دهید.
02:52	.خروجی نمایش داده می‌شود.
02:54	حالا طرز کار <code>while loop</code> را می‌بینیم.
02:57	پنجره را تغییر اندازه می‌دهیم.
03:00	در ابتدا مقدار <code>x</code> و <code>y</code> صفر (0) می‌باشد.
03:04	این وضعیت <code>while</code> ما می‌باشد.
03:06	در اینجا ما بررسی می‌کنیم که <code>x</code> کمتر و یا مساوی 10 می‌باشد یعنی مقدار <code>x</code> از 0 تا 10 می‌باشد.
03:15	سپس ما <code>y</code> و <code>x</code> را جمع می‌کنیم. یعنی 0 به اضافه 0 و جواب 0 می‌باشد.
03:22	مقدار <code>y</code> را چاپ می‌کنیم که 0 می‌باشد.
03:27	حال <code>x</code> افزایش می‌یابد یعنی مقدار <code>x</code> یک (1) خواهد بود.

03:33	دوباره وضعیت را بررسی می کنیم. 1 کمتر و یا مساوی 10 میباشد. اگر وضعیت true (درست) باشد سپس ما مقادیر را جمع می کنیم.
03:44	و y یعنی 0 به اضافه x که 1 است. 0 به اضافه 1 که 1 است.
03:50	ما مقدار را 1 چاپ می کنیم.
03:53	دوباره x افزایش می یابد.
03:55	حالا مقدار x که 2 می باشد.
03:59	دوباره وضعیت را بررسی می کنیم.
04:01	2 کمتر از و یا مساوی 2 است. اگر وضعیت true (درست) باشد ما مقادیر را جمع می کنیم. 1 به اضافه 2 که 3 را می دهد.
04:11	مقدار را 3 چاپ می کنیم.
04:13	همین طور تا جایی که x کمتر و یا مساوی 10 باشد ($x \leq 10$) ادامه پیدا می کند.
04:20	حالا همین برنامه را با استفاده از do....while loop می بینیم.
04:24	در اینجا برنامه ما می باشد.
04:26	نام فایل ما do hyphen while dot c می باشد.
04:31	این قسمت در برنامه قبلی توضیح داده شده است.
04:35	به قسمت do....while loop می رویم.
04:38	ابتدا بدنه حلقه اجرا می شود و سپس وضعیت بررسی می شود.
04:44	مقدار x به مقدار y اضافه می شود و حاصل جمع در y ذخیره می شود.
04:52	مفهوم مثل همان برنامه while می باشد.
04:55	حالا برنامه را اجرا می کنیم.
04:58	به ترمینال باز می گردیم.
05:00	و gcc space do hyphen while dot c space hyphen o space do Enter را فشار دهید.
05:08	و ./do را تایپ کنید. Enter را فشار دهید.
05:12	خروجی مثل همان در برنامه while می باشد.
05:16	حالا طرز کار do....while loop را می بینیم.
05:20	پنجره را تغییر اندازه می دهیم.
05:22	مقدار x و y صفر (0) می باشد.

05:25	این مقادیر را جمع می‌کنیم و 0 را بدست می‌آوریم.
05:29	حالا مقدار y صفر (0) می‌باشد.
05:31	ما مقدار 0 را چاپ می‌کنیم.
05:33	سپس x با مقدار 1 افزایش می‌یابد یعنی حالا مقدار x یک (1) می‌باشد. وضعیت بررسی می‌شود.
05:42	می‌بینید که بدنه حلقه ابتدا اجرا می‌شود.
05:45	اگر وضعیت <code>false</code> (اشتباه) باشد ما مقدار 0 را بدست می‌آوریم.
05:52	حالا بررسی می‌کنیم که 1 کمتر و یا مساوی 10 می‌باشد.
05:56	وضعیت درست می‌باشد دوباره مقادیر را با هم جمع می‌کنیم.
06:00	حالا 0 به اضافه 1
06:02	مقدار y را 1 چاپ می‌کنیم.
06:05	دوباره x افزایش می‌یابد.
06:08	حالا مقدار x دو (2) می‌باشد.
06:11	حالا بررسی می‌کنیم که 2 کمتر و یا مساوی 10 می‌باشد.
06:15	ما به اینجا بر می‌گردیم.
06:17	سپس مقادیر را با هم جمع می‌کنیم. 1 به اضافه 2 که 3 می‌باشد.
06:20	مقدار y را 3 چاپ می‌کنیم.
06:23	وضعیت تا زمانی که مقدار x کمتر و یا مساوی 10 باشد بررسی می‌شود.
06:30	و این دستور <code>return</code> ما می‌باشد.
06:33	در اینجا وضعیت <code>while</code> با <code>semicolon</code> پایان می‌یابد.
06:38	در <code>while loop</code> وضعیت با <code>semicolon</code> پایان نمی‌یابد.
06:43	حالا ببینیم این برنامه را چگونه در <code>C++</code> اجرا کنیم.
06:48	این برنامه <code>while</code> ما در <code>C++</code> می‌باشد.
06:52	مفهوم و پیاده‌سازی مثل همان برنامه <code>C</code> می‌باشد.
06:56	تعدادی تغییرات مثل <code>header file</code> که <code>iostream</code> بجای <code>stdio.h</code> می‌باشد.
07:04	در اینجا دستور <code>using</code> را شامل کرده‌ایم <code>using namespace std</code> و اینجا عملگر <code>cout</code> را بجای <code>printf</code> داریم.
07:16	ساختار <code>while loop</code> مثل همان در برنامه <code>C</code> می‌باشد.
07:21	حالا برنامه را اجرا می‌کنیم.

07:23	به ترمینال بر می گردیم.
07:25	Prompt را پاک می کنم.
07:28	برای اجرا <code>g++ space while dot cpp space hyphen o space while1</code> را تایپ کنید. Enter را فشار دهید.
07:38	و <code>./while1</code> را تایپ کنید. Enter را فشار دهید.
07:43	می بینید که خروجی مثل همان برنامه <code>while</code> در C می باشد.
07:48	حالا برنامه <code>do... while</code> را در C++ می بینیم.
07:52	به editor متنی بر می گردیم.
07:54	در اینجا نیز تغییرات مشابه مثل <code>header file</code> و <code>using statement</code> و عملگر <code>cout</code> می باشد.
08:03	بقیه مثل قبل می باشد.
08:06	برنامه را اجرا می کنیم.
08:08	به ترمینال بر می گردیم.
08:10	و <code>g++ space do hyphen while dot cpp space hyphen o space do1</code> را تایپ کنید. Enter را فشار دهید.
08:19	و <code>./do1</code> را تایپ کنید. Enter را فشار دهید.
08:23	خروجی مثل همان <code>do... while program in C</code> می باشد.
08:28	حالا تعدادی اشتباهات رایج و راه حل آن را بررسی می کنیم.
08:32	به editor متنی بر می گردیم.
08:35	فرض کنید اینجا من مقدار <code>x</code> را افزایش نمی دهم.
08:41	Save را کلیک کنید. ببینید چه می شود.
08:44	به ترمینال برگردید. Prompt را واضح می کنم.
08:47	برنامه را اجرا می کنیم.
08:50	کلید فلش بالا را دو بار فشار دهید.
08:54	دوباره کلید فلش بالا را فشار دهید.
08:57	خروجی نمایش داده می شود.
08:59	ما تعدادی صفر می بینیم چون حلقه (<code>loop</code>) وضعیت پایان دهنده ندارد.
09:07	و این <code>infinite loop</code> نامیده می شود.
09:10	<code>Infinite loop</code> باعث می شود که سیستم کار نکند.

09:14	و باعث می‌شود که برنامه همه <code>processors time</code> را مصرف کند ولی آن را می‌شود پایان داد.
09:21	به برنامه برمی‌گردیم و اشتباه را تصحیح می‌کنیم.
09:25	و <code>x++</code> و <code>semicolon</code> را تایپ کنید.
09:28	<code>Save</code> را کلیک کنید. دوباره اجرا می‌کنیم.
09:31	به ترمینال باز می‌گردیم.
09:33	کلید فلش بالا را فشار دهید.
09:38	بله. کار می‌کند.
09:40	این ما را به پایان این برنامه می‌آورد.
09:43	به اسلایدهایمان برمی‌گردیم.
09:45	خلاصه می‌کنم.
09:47	این موارد را یاد گرفتیم.
09:50	<code>While loop</code> مثل (<code>x</code> کمتر و یا مساوی 10) <code>while</code>
09:54	و <code>do.... While loop</code>
09:56	مثل <code>do statement block</code> و
09:59	وضعیت <code>while</code> در آخر
10:01	ارائه:
10:03	یک برنامه برای چاپ 0 تا 9 با استفاده از <code>for loop</code> بنویسید.
10:07	
10:10	ساختار <code>for loop</code> این‌گونه است:
10:12	<code>for (variable initialization; variable condition; and variable increment or decrement)</code>
10:20	و اینجا بدنه حلقه می‌باشد.
10:24	ویدیو را در لینک زیر مشاهده کنید.
10:27	این خلاصه پروژه <code>spoken tutorial</code> می‌باشد
10:30	اگر پهنای باند خوبی ندارید، ابتدا دانلود سپس مشاهده کنید
10:33	تیم پروژه <code>Spoken Tutorial</code> کارگاه آموزشی استفاده از <code>spoken tutorial</code> ارائه میدهد
10:35	
10:38	و به کسانی که آزمون آنلاین را قبول شوند گواهینامه میدهد

10:42	برای جزئیات بیشتر لطفا به "contact@spoken-tutorial.org" ایمیل بفرستید
10:47	Spoken tutorial بخشی از پروژه Talk To a Teacher می باشد.
10:51	که تحت پشتیبانی National Mission on Education از طریق ICT, توسط MHRD دولت هند می باشد.
10:58	اطلاعات بیشتر در لینک زیر موجود می باشد
11:02	ترجمه و صداگذاری توسط شبلم اقبال خان..
11:08	با تشکر از شرکت شما