

Exemples:

Ce va afișa codul de mai jos? <pre>int f(int x) { while(x < 20) { x *= 2; } return x; } ... int x=3, y; y = f(x); printf("%d %d", x, y);</pre>		Ce semnificație are declararea unei variabile locale unei funcții folosind prefixarea "static"? <pre>int func(){ static int max=25; return --max; } printf("%d", func()); printf("%d", func()); printf("%d", func());</pre>	
A	24 24	A	Variabila nu mai poate fi modificată
B	3 24	B	Variabila devine vizibilă și în afara funcției
C	3 12	C	Se inițializează o singură dată, indiferent de numărul de apeluri ale funcției
D	3 20	D	Nici o semnificație

int n = 5; $n / 2 * 10 = ?$		Expresia $x \& (x - 1)$ are valoarea 0 dacă:	
A	2.5	A	x este putere a lui 2
B	25	B	x nu este putere a lui 2
C	0.25	C	x este par
D	20	D	niciodată

<p>What will be displayed as result?</p> <pre> int f(int x) { while(x < 20) { x *= 2; } return x; } ... int x=3, y; y = f(x); printf("%d %d", x, y); </pre>		<p>What is the meaning of prefix „static” when declaring a local variable to a function, like in the example?</p> <pre> int func(){ static int max=25; return --max; } printf("%d", func()); printf("%d", func()); printf("%d", func()); </pre>	
A	24 24	A	The variable can't be modified
B	3 24	B	The variable becomes visible outside the function
C	3 12	C	Is initialized once, regardless of the number of function calls
D	3 20	D	No special meaning

What is the result printed?		What is the value of x after running the code?	
<pre> #include <stdio.h> int main(){ int sir[5]={1,2,3,4,5}, i; int *p; p=sir+2; *p=10; for(i=0;i<4;i++) printf("%d, ", sir[i]); printf("%d\n",sir[4]); return 0; } </pre>		<pre> void sum(int n1, int n2, int *n3) { n3 = n1 + n2; } ... int x = 0; sum(1, 2, &x); </pre>	
A	No result printed	A	0
B	Compile error	B	1
C	1, 2, 3, 4, 5	C	2
D	1, 2, 10, 4, 5	D	3