

CS 301 Midterm Exam

0.) NAME: _____

2015-10-21. Closed book, closed laptop, closed notes.

1.) Fill in the blanks to make true statements, for a 64-bit Linux machine (like NetRun's default).

- A function's return value is stored _____.
- A function's first parameter is stored _____.
- A class with one long uses _____ bytes of memory.
- You would write a C++ prototype for an assembly function named “bar” with no parameters or return value as:

(Score: 10 points.)

2.) Convert each piece of C++ code to assembly, and vice versa:

C++	Assembly
	<pre>grabit: mov rax, 5 ret</pre>
<pre>// The functions phase1 and phase2 // both take and return one long. long pipeline(long x) { return phase2(phase1(x)+3); }</pre>	

(Score: 30 points. Each piece of code is separate.)

3.) There are several memory allocation styles. Fill in the remaining table entries.

What?	Why?	How? (allocate one long)	Static or Dynamic?
section .data		dq 3	
the stack			
malloc			

(Score: 10 points. “Static” allocation happens once per program; “Dynamic” can happen again and again.)

4.) Fix the errors in this assembly translation of this C++: `for (int i=0;i<n;i++) arr[i]=3;`

```

; input: rsi == arr, rdi == n
mov rcx,0 ; i
start:
mov DWORD[rsi + rcx], 3
add rcx,1
cmp rcx,rdi
jle start

```

(Score: 15 points.)

5.) You see the compiler uses “push rbx” at the start of your function, and “pop rbx” at the end. Why?

(Score: 10 points.)

6.) If each piece of code returns a value, write the value. If it'll crash or hang, write that, and why.

<pre> push rax push 3 pop rax ret </pre>	<pre> push 2 push 7 pop rax pop rcx ret </pre>	<pre> mov rax, 9 push rax sub rsp, 8 ret </pre>	<pre> push 6 mov rax,QWORD[rsi] pop rcx ret </pre>	<pre> mov rax,0x3F and rax,0x55 ret </pre>

(Score: 15 points. Several of these are ... subtle. Be careful.)

7.) You had asked your newest employee to measure the speed of an empty function on Intel's new server chip, and his entire email in response was “503,148.7 per second”. What's wrong with his response?

(Score: 10 points.)