Sample Questions
500083

Types

- How many bits are used to represent a double in C++
  1. 32
  2. 4
  3. 64
  4. 8
  5. None of the above

Answer = 3

Assembly

- Consider
  - mov eax, dword_ptr[x]

- What is the effect of this line?
  1. Save the contents (16 bits) of the eax register to variable x
  2. Load the value (16 bits) currently stored in variable x to the eax register
  3. Save the contents (32 bits) of the eax register to variable x
  4. Load the value (32 bits) currently stored in variable x to the eax register
  5. Find the address of variable x and load it into the eax register
  6. None of the above

Answer = 4

The stack

- Which of the following statements about the C++ stack are correct?
  1. A stack is an ordered collection of data, contiguous in memory
  2. A stack is an unordered collection of data
  3. On the x86 architecture it is safe to access data on the stack, both above and below the stack pointer
  4. On the x86 architecture, the stack grows upwards in memory
  5. The base pointer, identifies the previous position of the stack pointer prior to the allocation of the current function's local variables
  6. All local variables are stored on the stack
  7. The assembly "call" operator places the address of the called function on the stack

Select all that apply. Marks are lost for wrong answers

Answer = 1, 5, 6,

Pointers

- Which of the following statements on C++ pointers is INCORRECT?
  1. A pointer represents the address of a location in memory
  2. De-referencing a pointer gains access to the data at the address stored in the pointer.
  3. Taking the address of any variable, results in a pointer
  4. Adding an integer to a pointer is a meaningful operation
  5. Multiplying a pointer by an integer is a meaningful operation

Answer = 5

Parameter passing

- Which of the following statements about "by-value" are correct?
  1. Changes to "pass-by-value" parameters within the called function effect the parameters in the calling function.
  2. "Pass-by-value" parameters are copied onto the stack
  3. "Pass-by-value" parameters are transferred to the called function in the EAX register
  4. Data returned from a function using "return-by-value" is copied onto the stack
  5. Data returned from a function using "return-by-value" is transferred to the calling function in the EAX register

Select all that apply. Marks are lost for wrong answers

Answers = 2, 5

References and Pointers

- What is the difference between a reference and a pointer, in C++?
  1. A reference stores the value of a variable, whereas a pointer stores the address of a variable.
  2. A pointer once assigned an address cannot be altered, whereas a reference can be assigned multiple addresses.
  3. The following are considered functionally identical:
     - const int *p;
     - int &r;
  4. A reference can reference a non-existent object, whereas a pointer would have to store the NULL value.
  5. None of the above.

Answer = 5

Parameter passing

- Given the following C++ function prototype:
  - void theResult (int length, int& width);
- Which of the following lines of assembly, best describe how the function is called

1.
```
lea     eax, [width]
push    eax
lea     eax, [length]
push    eax
call    theResult (445670h)
```

2.
```
lea     eax, [length]
push    eax
mov     ecx, dword ptr [width]
push    ecx
call    theResult (445670h)
```

3.
```
lea     eax, [width]
push    eax
mov     eax, dword ptr [length]
push    eax
call    theResult (445670h)
```

4.
```
lea     eax, [width]
push    eax
mov     ecx, word ptr [length]
push    ecx
call    theResult (445670h)
```

Answer = 3

Arrays and Pointers

- Given the C++ code:

```
1. int a[100];
2. int *b = a;
3. int *c = &a[10];

4. for (int i=0; i<100; i++)
5.     b[i] = i;

6. for (int j=0; j<10; j++)
7.     c[j] = c[j] + j;

8. c = c + 2;

9. for (int j=0; j<10; j++)
10.    c[j] = c[j] + j;
```

- What is the value of a[12] ?

Answer = 14