CS 1021C-001 - Spring 2014 - Exam 2
University of Cincinnati

There are 6 questions for a total of 115 points. Your final score will be out of 100.

Name: _____

Instructions

- Please read through this entire exam very carefully before starting.
- This exam is closed book
  - You may use a single piece of standard 8.5inch x 11inch US letter paper. You may write whatever you like on all 6 sides of the paper, as long it contains your name somewhere. You must turn this paper in with your exam.
- All work must be written on the exam pages in order to be graded. Any scrap paper used, must be the scrap paper provided during the exam period.
- For programming questions: Please be accurate with your C++ syntax: this includes appropriate use of braces, semicolons, and the proper use of upper/lowercase letters.
- No electronic devices may be used during the exam: this includes (but is not limited to) calculators, phones, tablets, and computers.
- You have 55 minutes to complete the exam.

DON'T PANIC!

Question:	1	2	3	4	5	6	Total
Points:	30	20	10	20	15	20	115
Score:							

Multiple Choice and True/False

1. Circle the **best** response.

[2] (a) The conditional input for an `if` statement must evaluate to this type:

A. `int`
B. `char`
C. `float`
D. `bool`

Solution: D - bool

[2] (b) Which operator joins two conditionals with the **and** relationship?

A. `||`
B. `&&`
C. `!`
D. `&`
E. `|`

Solution: B - `&&`

[2] (c) What is printed out if this code is executed?

```
1 int x = 27;
2 if (x > 27)
3   cout << "pop";
4   cout << "POP";
```

A. `nothing`
B. `pop`
C. `popPOP`
D. `POP`

Solution: D - POP

[2] (d) It is illegal to use an assignment = as the input to a conditional (i.e. `if`):

A. true
B. false

Solution: B - false, legal but not what you want in most cases

[2] (e) The operand of the increment and decrement operators (++ and −−) can be any valid mathematical expression.

A. true
B. false

Solution: false

[2] (f) The `while` loop is a pretest loop

A. true
B. false

Solution: true

[2] (g) The `do` loop is a pretest loop

A. true
B. false

Solution: false

[2] (h) The `for` loop is a posttest loop

A. true
B. false

Solution: false

[2] (i) To calculate the total number of iterations of a nested loop, add the number of iterations of all the loops

A. true
B. false

Solution: false

[2] (j) The `while` loop is a pretest loop

A. true
B. false

Solution: true

[2] (k) The first element in an array has an index of

A. 0
B. 1
C. *size*
D. $size - 1$

Solution: A - zero

[2] (l) With this array declaration, `string strArry[10];` What is each element initialized to?

A. random / whatever is in memory
B. `""`
C. `"0"`

Solution: B - the empty string

[2] (m) For function calls in C++, the default type of parameter passing is:

A. pass-by-value
B. pass-by-reference
C. pass-by-pointer

Solution: A - pass-by-value

[2] (n) Of the sorting algorithms we studied, wish does less swaps in general?

A. bubble sort
B. selection sort

Solution: B - selection sort

[2] (o) All pointer variables are the same size in memory:

A. true
B. false

Solution: true

Fill in the blank

2. Fill in the blanks with the correct answer.

[4] (a) The _____ and _____ loops will not iterate at all if their test expressions are false to start with.

Solution: for, while

[2] (b) A loop that is inside another loop is call a _____ loop.

Solution: nested

[2] (c) The _____ statement causes a loop to skip the remaining statements in the current iteration.

Solution: continue

[2] (d) A return type of _____ indicates that a function does not return a value.

Solution: `void`

[2] (e) Having two functions with the same name, but different parameter lists is called _____.

Solution: overloading

[2] (f) The _____ search algorithm repeatedly divides the portion of an array being searched in half.

Solution: binary

[2] (g) _____ variables are designed to hold memory addresses.

Solution: pointer

[2] (h) To access the value in memory pointed to by a pointer variable, you must _____ the pointers.

Solution: dereference

[2] (i) _____ is used to signify that a pointer doesn't point to anything.

Solution: NULL

Short Answer

3. Write a brief response to the questions below. You shouldn't need more space than provided.

[5] (a) So far, we have learned three different uses for the * operator. What are they?

Solution:

- multiplication
- declare a pointer
- dereference a pointer

[5] (b) Describe how a two dimensional array is actually stored in memory.

Solution: A two dimensional array is stored in memory as a one dimensional array. Where the rows are stored back to back with each having the full number of columns.

Code Analysis

20 4. Examine the code below for a sorting algorithm (a modified selection sort).

```
1  void someKindOfSort(int arr[], int size) {
2    int rounds = size / 2 + 1;
3    for (int it = 0; it < rounds; it++) { // Will be sorted in size/2 rounds
4      int pMi = it;                  // index of smallest value
5      int upperId = size - it - 1;
6      int pMa = upperId;             // index of largest value
7      if (it >= upperId) {
8        continue;
9      }
10     // Find the index of the smallest and largest
11     for(int idx = it; idx < size - it; idx++) {
12       if (arr[idx] < arr[pMi]) {
13         pMi = idx;
14       }
15       if (arr[idx] > arr[pMa]) {
16         pMa = idx;
17       }
18     }
19     // Do some swaps. I hope this is right.
20     if (pMi != it) {
21       swap(arr, it, pMi);
22       if (pMa == it) {
23         pMa = pMi;
24       }
25     }
26     if (pMa != upperId) {
27       swap(arr, upperId, pMa);
28     }
29     print(arr, size);
30   }
31 }
32
33 int main() {
34   int arr[SIZE] = {10, 1, 9, 2, 8, 3, 7, 4, 6, 5};
35   print(arr, SIZE);
36   cout << "--sort--" << endl;
37   someKindOfSort(arr, SIZE);
38   cout << "--final--" << endl;
39   print(arr, SIZE);
40   return 0;
41 }
```

What is the output of this program. You should assume the existence of the `print` method that prints out the contents of the array in order and the existence of the `swap` method that correctly swaps two value in the array. I have provided the first 2 and the last 2 lines of the output. Here, more than enough space is provided, don't assume you need to fill it.

```
 10  1  9  2  8  3  7  4  6  5
--sort--
```

Solution:

```
 1  5  9  2  8  3  7  4  6 10
 1  2  6  5  8  3  7  4  9 10
 1  2  3  5  4  6  7  8  9 10
 1  2  3  4  5  6  7  8  9 10
 1  2  3  4  5  6  7  8  9 10

--final--
  1  2  3  4  5  6  7  8  9 10
```

Programming Questions

15 5. Write a function that takes in a pointer to an array of integers, and the size of that array. Your function should create a dynamic array of the same size and return an array that contains the same elements as the array passed in, except they will be in reverse order.

```
// Allocates a new array and populates it with the elements of arr,
// but in reverse order.
int* reverseArray(int *arr, int size) {
  // BEGIN ANSWER
```

Solution:

```
int* rtn = new int[size];
for (int i = 0; i < size; i++) {
  rtn[size - i - 1] = arr[i]
}
return rtn;
```

```
  // END ANSWER
}
```

20 6. When dealing with a two dimensional matrix A, the transpose operation (A^T) is the result of flipping the matrix over the main diagonal. For example, see the original matrix on the left and the transpose of that matrix on the right. Write the necessary code to transpose a square matrix stored in a two dimensional array. The transpose should be done "in place" meaning that a new array is not allocated and the data isn't fully copied.

A						A^T				
0	1	2	3	4		0	5	10	15	20
5	6	7	8	9		1	6	11	16	21
10	11	12	13	14		2	7	12	17	22
15	16	17	18	19		3	8	13	18	23
20	21	22	23	24		4	9	14	19	24

```
// it doesn't matter what SIZE actually is.
const int SIZE = ?;
// the matrix is always square
int arr[SIZE][SIZE];
int counter = 0;
for (int r = 0; r < SIZE; r++) {
  for (int c = 0; c < SIZE; c++) {
    arr[r][c] = counter++;
  }
}
// BEGIN ANSWER - transpose arr
```

Solution:

```
for (int r = 0; r < SIZE; r++) {
  for (int c = r + 1; c < SIZE; c++) {
    int tmp = arr[r][c];
    arr[r][c] = arr[c][r];
    arr[c][r] = tmp;
  }
}
```

```
// END ANSWER - transpose arr
```

THIS PAGE LEFT UNINTENTIONALLY BLANK