CSCE240 Spring 2017 SI: Exam Review 1 Feb 1, 2017

Write the output of the code.

1
```
for (int i = 0; i < 5; i++) {
    for (int j = 0; j < i; j++) {
        cout << "*";
    }
    cout << endl;
}
```

2
```
for (int i = 0, j = 10; i < j;
i++, j--) {
    cout << i << " " << j << endl;
}
```

3
```
int i = 0;
int j = 7;
while (j != 0) {
    for (int k = j - i; k >= 0; k-
-) {
        cout << "*";
    }
    cout << endl;
    i++;
    j--;
}
```

4
```
int a[] = {2,5,4,3,1,0,6,2};
int a_size = 8;
int b = 0;
for (int i = 0; i < a_size; i++) {
    for (int j = i; j > 0; j--) {
        if (a[j] < a[j - 1]) {
            b = a[j];
            a[j] = a[j - 1];
            a[j - 1] = b;
        }
    }
}
b = a_size - 1;
do {
    cout << a[b] << " ";
} while (b-- > 0);
cout << endl;
```

5
```
int a[5][5];
int a_width = 5;
int a_height = 5;
for (int i = 0; i < a_width; i++)
{
    for (int j = 0; j < a_height;
j++) {
        a[i][j] = (i * j) + 1;
    }
}
for (int i = 0; i < a_width; i++)
{
    for (int j = 0; j < a_height;
j++) {
        cout << a[i][j] << " ";
```

```
        }
        cout << endl;
    }
```

6

```
    int * a, * b;
    int N = 5;
    a = new int[N];
    b = a;
    for (int i = 0; i < N; i++) {
        a[i] = i * N;
    }
    cout << *++a << endl;
    delete [] b;
```

7

```
    int a[] = {0,1,2,3,4,5,6,7,8,9};
    int a_size = 10;
    int * p;
    int b = a_size - 1;
    p = a;
    while (b >= 0) {
        cout << p[b] << " ";
        b--;
    }
    cout << endl;
```

8

```
    int a[] = {0,1,2,3,6,5,4,9,7};
    int b = 1;
    cout << (b += 30) << endl;//31
    cout << (++b -= 21) << endl;//10
    cout << (--b %= 4) << endl;

    cout << a[b] << endl;
```

9

```
    int n = 15;
    cout << (n++) << endl;
    cout << (++n) << endl;
    cout << "(n++)" << endl;
```

10

```
    char str1[] = "The Cake Is a Lie";
    char str2[] = "Wubba Lubba Dub
    Dub";
    char str3[] =
    {'s','t','a','y','i','n','g','\0',
    'a','l','i','v','e','\0'};
    char * cat;
    int a = 0;
    int b = 0;
    int c = 0;
    int d = 0;
    cat = new char[strlen(str1) +
    strlen(str2) + strlen(str3)];
    cout << str1 << endl << str2 <<
    endl << str3 << endl;
    while (str1[a] != '\0' || str2[b]
    != '\0' || str3[c] != '\0') {
        if (str1[a] != '\0') {
            cat[d] = str1[a];
            d++;
            a++;
        }
        if (str2[b] != '\0') {
```

```
            cat[d] = str2[b];
            d++;
            b++;
        }
        if (str3[c] != '\0') {
            cat[d] = str3[c];
            d++;
            c++;
        }
    }
    for (int i = 0; i < strlen(cat);
    i++) {
        cout << cat[i];
    }
    cout << endl;
```

11

```
    int x = 2;
    switch (x) {
        case 1:
            cout << "Case " << 1 <<
    endl;
        case 2:
            cout << "Case " << 2 <<
    endl;
        case 3:
            cout << "Case " << 3 <<
    endl;
            break;
        case 4:
            cout << "Case " << 4 <<
    endl;
        default:
            cout << "Default Case" <<
    endl;
    }
```

12

```
    int a = 8;
    int b = 4;
    int c;
    if ((c = b) == 4) {
        cout << true << endl;
        c = a / 2;
    }
    if (c++ > ++b) {
        cout << "yes" << endl;
    } else {
        cout << c << endl;
    }
    while (++b < 10) {
        cout << b << endl;
        if (b > a) break;
    }
```

13

```
    char a[] = "Dan the dude";
    char b[] = "Thank You MArio, but
    our princess is in another
    castle";
    char c[] = {'3','l','1','t','3'};
    char f[strlen(a)];
    char * d = a;
```

```
char * e = f;
int i = 0;
while (( *e = *d) != '\0') {
    d++;
    e++;
}
cout << f << endl;
d = new char[strlen(a)];
strcpy(a,d);
cout << strlen(d) << endl;
while (b[i] != '\0' && c[i] !=
'\0') {
    cout << b[i] << c[i];
    i++;
}
cout << endl;
```

Find the Errors in the Code and describe what the program behavior would be (such as a Compiler Error, Array out of bounds, etc.)

1
```
int a[] = new int[10];
int sum = 0;
for (int i = 0; i < 10; i++) {
    a[i] = i + sum;
    sum = a[i];
}
cout << sum << endl;
```

2
```
char a[] = "Its-a-me Mario!";
char * b;
strcpy(b, a);
cout << b << endl;
```

3
```
int a[5];
for (int i = 15; i >= 0; i--) {
    a[i - 1] = i;
}
cout << a[6] << endl;
```

4
```
int a = 1;
switch (a) {
    case a > 0:
        cout << "A is Greater than
0" << endl;
    case a == 0:
        cout << "A is Zero" <<
endl;
    case a < 0:
        cout << "A is Less than
Zero" << endl;
}
```

5
```
char a[] =
{'T','a','n','g','e','r','i','n','
e'};
char *b = new char[strlen(a)];
while ((*b = *a) != '\0') {
    b++;
    a++;
}
cout << b << endl;
```

6
```
int a = -1337;
if (a = 1) {
    cout << "A is 1" << endl;
}
```

7
```
float a =
9222524.024659756321457753215465135462168432;
float b =
9222524.024659756321457753215465135462168432;
if (a == b) {
    cout << "A and B are the same"
<< endl;
}
```

8
```
int i = 0;
cout << i++ << " " << ++i << endl;
```

9
```
const int hello = 42;
cout << hello++ << endl;
```

10
```
int a[10];
int * b;
for (int i = 0; i < 10; i++) {
    b[i] = a[i];
}
```

11
```
int a[] = {0,1,2,3};
int b[4];
b = a;
cout << b << endl;
```

12
```
int a[10];
int *b = new int[10];
for (int i = 0; i < 10; i++) {
    b[i] = a[i];
    cout << b[i] << endl;
}
```

13
```
int a[3];
int * b = new int [3];
for (int i = 0; i++ < 3;) {
    b[i] = a[i];
}
delete b[];
cout << b[2] << endl;
```

14
```
int * a = new int[10];
for (int i = 0; i < 10; i++) {
    a[i] = i * 3;
    cout << a[i] << endl;
}
delete a;
```

You can find all of this code on the SI github:
https://github.com/Nesdood007/CSCE240S2017/tree/master/Worksheets/ExamReview1

To run this on the linux lab computers, open a terminal or SSH in and type

$ git clone https://github.com/Nesdood007/CSCE240S2017/

The repository should be cloned to ~/git/CSCE240S2017/

Given CStrings a and b, finish the function to concatenate the two strings. Make sure to return you concatenated CString!

```
char* concatenate (const char * a, const char * b) {
```

Write a program that prints, given some user input n, the following pattern

1. There are n lines in total
2. Every other line starting from the first line has the string “()” n times
3. Every other line starting from the second line has “(“ n times followed by “)” n times

```
//Example if n = 4:
//()()()()
//(((())))
//()()()()
//(((())))
```