

San José State University
Department of Computer Engineering

CMPE 180-92
Data Structures and Algorithms in C++
Spring 2017

Instructor: Ron Mak

Assignment #3B

Assigned: Thursday, September 7

Due: Thursday, September 14 at 5:30 PM

CodeCheck: <http://codecheck.it/codecheck/files/17011724027ahzerajkrzg2kd682p363d4>

Canvas: Assignment 3.b. Spirals

Points: 100

Spirals

This assignment will give you practice using a two-dimensional array that you pass to a function. Write a C++ program that arranges consecutive integers in a square spiral pattern which has the same number of rows and columns. At the above CodeCheck URL, complete the program **spirals.cpp**.

A spiral

A size-n spiral is an arrangement of consecutive integers starting at a given value within an n-by-n region of a matrix. Place the starting number in the middle of the region, and arrange the remaining numbers in an outward-growing counter-clockwise spiral. For example, see the size 5 spiral in the expected output below.

The size of a spiral can be 1 through 101 inclusive, and the size will always be odd. The starting value can be 1 through 50. Be sure to check the size and starting value. You will generate and print each matrix in the upper-left corner region of a fixed-size 101-by-101 integer matrix. For this assignment, CodeCheck will verify that you can generate and print the spirals in this table.

Size	Starting value
1	1
5	1
9	11
12 (invalid – not odd)	13
15	17

Your program must pass the matrix as a parameter to at least two functions, **make_spiral** and **print_spiral**.

Expected output

CodeCheck will verify that your program's results match this output exactly.

```
Spiral of size 1 starting at 1
```

```
1
```

```
Spiral of size 5 starting at 1
```

```
17 16 15 14 13
18  5  4  3 12
19  6  1  2 11
20  7  8  9 10
21 22 23 24 25
```

```
Spiral of size 9 starting at 11
```

```
75 74 73 72 71 70 69 68 67
76 47 46 45 44 43 42 41 66
77 48 27 26 25 24 23 40 65
78 49 28 15 14 13 22 39 64
79 50 29 16 11 12 21 38 63
80 51 30 17 18 19 20 37 62
81 52 31 32 33 34 35 36 61
82 53 54 55 56 57 58 59 60
83 84 85 86 87 88 89 90 91
```

```
Spiral of size 12 starting at 13
***** Error: Size 12 must be odd.
```

```
Spiral of size 15 starting at 17
```

```
213 212 211 210 209 208 207 206 205 204 203 202 201 200 199
214 161 160 159 158 157 156 155 154 153 152 151 150 149 198
215 162 117 116 115 114 113 112 111 110 109 108 107 148 197
216 163 118  81  80  79  78  77  76  75  74  73 106 147 196
217 164 119  82  53  52  51  50  49  48  47  72 105 146 195
218 165 120  83  54  33  32  31  30  29  46  71 104 145 194
219 166 121  84  55  34  21  20  19  28  45  70 103 144 193
220 167 122  85  56  35  22  17  18  27  44  69 102 143 192
221 168 123  86  57  36  23  24  25  26  43  68 101 142 191
222 169 124  87  58  37  38  39  40  41  42  67 100 141 190
223 170 125  88  59  60  61  62  63  64  65  66  99 140 189
224 171 126  89  90  91  92  93  94  95  96  97  98 139 188
225 172 127 128 129 130 131 132 133 134 135 136 137 138 187
226 173 174 175 176 177 178 179 180 181 182 183 184 185 186
227 228 229 230 231 232 233 234 235 236 237 238 239 240 241
```

Submission into Canvas

When you're satisfied with your program in CodeCheck, click the "Download" link at the very bottom of the Report screen to download a signed zip file of your solution. Submit this zip file into Canvas. You can submit as many times as you want until the deadline, and the number of submissions will not affect your score. Only your last submission will be graded.

Submit the signed zip file from CodeCheck into Canvas: **Assignment 3.b. Spirals**

Note: You must submit the signed zip file that you download from CodeCheck, or your submission will not be graded. Do not rename the zip file.

Rubric

Your program will be graded according to these criteria:

Criteria	Maximum points
Good output (as determined by CodeCheck) <ul style="list-style-type: none">• Correct output values.• Correct spiral format.	30 <ul style="list-style-type: none">• 10• 20
Good program design <ul style="list-style-type: none">• Good use of a two-dimensional array.• Size and starting values checked.• Array properly passed to functions <code>make_spiral</code> and <code>print_spiral</code>.• Good implementation of a spiral algorithm.	50 <ul style="list-style-type: none">• 10• 10• 10• 20
Good program style <ul style="list-style-type: none">• Descriptive variable names.• Meaningful comments.• Follow the coding style (formatting, braces, indentation, function declarations before the main, etc.) of the Savitch textbook.	20 <ul style="list-style-type: none">• 5• 5• 10

Academic integrity

You may study together and discuss the assignments, but what you turn in must be your individual work. Assignment submissions will be checked for plagiarism using Moss (<http://theory.stanford.edu/~aiken/moss/>). **Copying another student's program or sharing your program is a violation of academic integrity.** Moss is not fooled by renaming variables, reformatting source code, or re-ordering functions.

Violators of academic integrity will suffer severe sanctions, including academic probation. Students who are on academic probation are not eligible for work as instructional assistants in the university or for internships at local companies.