Consolidated Assignment 3 Report

This report contains the graded results for the newest of each exercise submitted to the assignment checker prior to 4/22/2020 1:40:31 PM PDT.

Student Name: Shaun Chemplavil
Student ID: U08713628
Contact e-mail: shaun.chemplavil@gmail.com
C/C++ Programming I (Section 146359)

```
Submitted:
   Exercise 0: 4/7/2020 4:47:04 PM PDT
   Exercise 1: 4/7/2020 7:43:46 PM PDT
   Exercise 2: 4/18/2020 8:01:12 AM PDT
   Exercise 3: 4/18/2020 8:00:44 AM PDT
```

Score (out of 20 possible): 20

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL. For help please contact the instructor at the email address provided on the "Announcements" page of the course website. The assignment checker DOES NOT GRADE your submissions but merely reports on issues so you can correct them and resubmit, thereby avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after the assignment deadline based solely upon the NEWEST submission of each exercise. BE WARY of correcting minor issues after the deadline because a late deduction will usually be much greater than a minor issue deduction.

```
   From:  <mailto:shaun.chemplavil@gmail.com>
   Subject: C1A3E0_U08713628
   Submitted: 4/7/2020 4:47:04 PM PDT
   Course: C/C++ Programming I (Section 146359)
   Student's name: Shaun Chemplavil
   Contact email: shaun.chemplavil@gmail.com
   Student ID: U08713628
   Assignment 3, Exercise 0
   Exercise point value: 6
   File submitted:
      C1A3E0_Quiz.txt
```

NOTE: The assignment checker does not check the correctness of quiz answers for this assignment.

Your submission has been accepted and will be graded manually by the instructor. You may resubmit it as many times as you wish before the assignment deadline. BE WARY of correcting minor issues after the deadline because a late deduction will usually be much greater than a minor issue deduction.

All Correct!

```
Shaun Chemplavil U08713628
shaun.chemplavil@gmail.com
C/C++ Programming I : Fundamental Programming Concepts
146359 Raymond L. Mitchell, Jr., M.S.
04/07/2020
C1A3E0_Quiz.txt
Answers to Quiz 3

1. D
2. B
3. B
4. C
5. E
6. B
```

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL. For help please contact the instructor at the email address provided on the "Announcements" page of the course website. The assignment checker DOES NOT GRADE your submissions but merely reports on issues so you can correct them and resubmit, thereby avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after the assignment deadline based solely upon the NEWEST submission of each exercise. BE WARY of correcting minor issues after the deadline because a late deduction will usually be much greater than a minor issue deduction.

```
   From:  <mailto:shaun.chemplavil@gmail.com>
   Subject: C1A3E1_U08713628
   Submitted: 4/7/2020 7:43:46 PM PDT
   Course: C/C++ Programming I (Section 146359)
   Student's name: Shaun Chemplavil
   Contact email: shaun.chemplavil@gmail.com
   Student ID: U08713628
   Assignment 3, Exercise 1
   Exercise point value: 3
   File submitted:
      C1A3E1_main.c

"Compile-time" results:
   No "compile-time" issues;
"Run-time" results:
   Program ran - No errors detected during preliminary testing (SEE ATTACHMENT);
```

```c
//
// Shaun Chemplavil U08713628
// shaun.chemplavil@gmail.com
// C / C++ Programming I : Fundamental Programming Concepts
// 146359 Raymond L. Mitchell Jr.
// 04 / 07 / 2020
// C1A3E1_main.c
// Win10
// Visual C++ 19.0
//
// This file will take a positive user integer input (input),
//  and display a table of the cubic sum to that number
//

#include <stdio.h>

int main(void)
{
   int input;
   short cubic_sum = 0;

   // Request and Store User Input
   printf("Enter integer value >=0: ");
   scanf("%d", &input);

   // Display Column Headers
   printf(
      "nbr   cubic sum\n"
      "---------------\n");

   // loop that controls current row display
   for (short nbr = 0; nbr <= input; ++nbr)
   {
      // Add cube of current number to cubic sum
      cubic_sum += (short)(nbr * nbr * nbr);

      // Print table contents for current row
      printf("%3hi %10hi\n", nbr, cubic_sum);
   }
   return 0;
}
```

```
********** C1 ASSIGNMENT 3 EXERCISE 1 AUTOMATIC PROGRAM RUN RESULTS **********

************* THE RESULTS BELOW HAVE BEEN PARTIALLY CHECKED AND **************
*************   NO ERRORS WERE FOUND.  HOWEVER, THIS DOES NOT   **************
*************  NECESSARILY MEAN THAT THERE ARE NO ERRORS.  THE  **************
*************   THE INSTRUCTOR WILL DO A MORE THOROUGH CHECK    **************
*************              DURING MANUAL GRADING.               **************


------------------------------ START OF 1ST RUN ------------------------------

Enter integer value >=0: 1
nbr   cubic sum
---------------
  0          0
  1          1


------------------------------- END OF 1ST RUN -------------------------------

-------------------- IMPLEMENTATION SPECIFICS FOR 2ND RUN --------------------
This test uses a 16-bit short, which overflows for cubic sums of values > 18.
The results for these values are not tested.
------------------------------ START OF 2ND RUN ------------------------------

Enter integer value >=0: 25
nbr   cubic sum
---------------
  0          0
  1          1
  2          9
  3         36
  4        100
  5        225
  6        441
  7        784
  8       1296
  9       2025
 10       3025
 11       4356
 12       6084
 13       8281
 14      11025
 15      14400
 16      18496
 17      23409
 18      29241
 19     -29436
 20     -21436
 21     -12175
 22      -1527
 23      10640
 24      24464
 25     -25447


------------------------------- END OF 2ND RUN -------------------------------

-------------------- IMPLEMENTATION SPECIFICS FOR 3RD RUN --------------------
This test uses a 16-bit short, which overflows for cubic sums of values > 18.
```

```
The results for these values are not tested.
------------------------------ START OF 3RD RUN ------------------------------

Enter integer value >=0: 36
nbr   cubic sum
---------------
  0          0
  1          1
  2          9
  3         36
  4        100
  5        225
  6        441
  7        784
  8       1296
  9       2025
 10       3025
 11       4356
 12       6084
 13       8281
 14      11025
 15      14400
 16      18496
 17      23409
 18      29241
 19     -29436
 20     -21436
 21     -12175
 22      -1527
 23      10640
 24      24464
 25     -25447
 26      -7871
 27      11812
 28     -31772
 29      -7383
 30      19617
 31     -16128
 32      16640
 33     -12959
 34      26345
 35       3684
 36     -15196


------------------------------- END OF 3RD RUN -------------------------------
```

THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL. For help please contact the instructor at the email address provided on the "Announcements" page of the course website. The assignment checker DOES NOT GRADE your submissions but merely reports on issues so you can correct them and resubmit, thereby avoiding unnecessary credit loss. ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after the assignment deadline based solely upon the NEWEST submission of each exercise. BE WARY of correcting minor issues after the deadline because a late deduction will usually be much greater than a minor issue deduction.

```
   From: Shaun Chemplavil <mailto:shaun.chemplavil@gmail.com>
   Subject: C1A3E2_U08713628
   Submitted: 4/18/2020 8:01:12 AM PDT
   Course: C/C++ Programming I (Section 146359)
   Student's name: Shaun Chemplavil
   Contact email: shaun.chemplavil@gmail.com
   Student ID: U08713628
   Assignment 3, Exercise 2
   Exercise point value: 5
   File submitted:
      C1A3E2_main.cpp

"Compile-time" results:
   No "compile-time" issues;
"Run-time" results:
   Program ran - No errors detected during preliminary testing (SEE ATTACHMENT);
```

```
//
// Shaun Chemplavil U08713628
// shaun.chemplavil@gmail.com
// C / C++ Programming I : Fundamental Programming Concepts
// 146359 Raymond L.Mitchell, Jr., M.S.
// 04 / 18 / 2020
// C1A3E2_main.cpp
// Win10
// Visual C++ 19.0
//
// This file will take user integer input (inValue), and the reverse of it
//

#include <iostream>
using namespace std;

// Declare and set RADIX (base number of decimal system)
const int RADIX = 10;

int main()
{
   // Declare User Input (inValue)
   int inValue;
   //  Flag to signal sign of input value
   bool isNeg;

   // Request and Store User Input
   cout << "Enter any decimal integer value: ";
   cin >> inValue;

   // Display Opening Phrase
   cout << "\"" << inValue << "\" in reverse is  \"";

   // Track Original Sign of input
   isNeg = inValue < 0;

   // Force to be positive
   if (isNeg)
   {
      inValue = -inValue;
   }


   do
   {
      // Print Current Least Significant Bit
      cout << inValue % RADIX;

      // Remove the LSB
      inValue /= RADIX;
   } while (inValue != 0);

   // Check if we need to display negative sign
   if (isNeg)
   {
      cout << "-";
   }

   cout << "\"\n";
   return 0;
}
```

```
********** C1 ASSIGNMENT 3 EXERCISE 2 AUTOMATIC PROGRAM RUN RESULTS **********

************* THE RESULTS BELOW HAVE BEEN PARTIALLY CHECKED AND **************
*************   NO ERRORS WERE FOUND.  HOWEVER, THIS DOES NOT   **************
*************  NECESSARILY MEAN THAT THERE ARE NO ERRORS.  THE  **************
*************   THE INSTRUCTOR WILL DO A MORE THOROUGH CHECK    **************
*************              DURING MANUAL GRADING.               **************


------------------------------ START OF 1ST RUN ------------------------------

Enter any decimal integer value: 3987
"3987" in reverse is  "7893"

------------------------------- END OF 1ST RUN -------------------------------

------------------------------ START OF 2ND RUN ------------------------------

Enter any decimal integer value: -2645
"-2645" in reverse is  "5462-"

------------------------------- END OF 2ND RUN -------------------------------

------------------------------ START OF 3RD RUN ------------------------------

Enter any decimal integer value: 100
"100" in reverse is  "001"

------------------------------- END OF 3RD RUN -------------------------------

------------------------------ START OF 4TH RUN ------------------------------

Enter any decimal integer value: 000120
"120" in reverse is  "021"

------------------------------- END OF 4TH RUN -------------------------------

------------------------------ START OF 5TH RUN ------------------------------

Enter any decimal integer value: -0023
"-23" in reverse is  "32-"

------------------------------- END OF 5TH RUN -------------------------------

------------------------------ START OF 6TH RUN ------------------------------

Enter any decimal integer value: 000
"0" in reverse is  "0"

------------------------------- END OF 6TH RUN -------------------------------

------------------------------ START OF 7TH RUN ------------------------------

Enter any decimal integer value: 1010
"1010" in reverse is  "0101"

------------------------------- END OF 7TH RUN -------------------------------

------------------------------ START OF 8TH RUN ------------------------------
```

```
Enter any decimal integer value: -1010
"-1010" in reverse is  "0101-"

------------------------------- END OF 8TH RUN -------------------------------

------------------------------ START OF 9TH RUN ------------------------------

Enter any decimal integer value: -0007000
"-7000" in reverse is  "0007-"

------------------------------- END OF 9TH RUN -------------------------------

------------------------------ START OF 10TH RUN ------------------------------

Enter any decimal integer value: 3
"3" in reverse is  "3"

------------------------------- END OF 10TH RUN -------------------------------
```

```
THIS WAS SENT FROM A NOTIFICATION-ONLY ADDRESS THAT CANNOT ACCEPT INCOMING MAIL.
For help please contact the instructor at the email address provided on the
"Announcements" page of the course website.  The assignment checker DOES NOT GRADE your
submissions but merely reports on issues so you can correct them and resubmit, thereby
avoiding unnecessary credit loss.  ALL GRADING IS DONE MANUALLY BY THE INSTRUCTOR after
the assignment deadline based solely upon the NEWEST submission of each exercise.  BE
WARY of correcting minor issues after the deadline because a late deduction will
usually be much greater than a minor issue deduction.


   From: Shaun Chemplavil <mailto:shaun.chemplavil@gmail.com>
   Subject: C1A3E3_U08713628
   Submitted: 4/18/2020 8:00:44 AM PDT
   Course: C/C++ Programming I (Section 146359)
   Student's name: Shaun Chemplavil
   Contact email: shaun.chemplavil@gmail.com
   Student ID: U08713628
   Assignment 3, Exercise 3
   Exercise point value: 6
   File submitted:
      C1A3E3_main.cpp

"Compile-time" results:
   No "compile-time" issues;
"Run-time" results:
   Program ran - No errors detected during preliminary testing (SEE ATTACHMENT);
```

```
//
// Shaun Chemplavil U08713628
// shaun.chemplavil@gmail.com
// C / C++ Programming I : Fundamental Programming Concepts
// 146359 Raymond L.Mitchell, Jr., M.S.
// 04 / 18 / 2020
// C1A3E3_main.cpp
// Win10
// Visual C++ 19.0
//
// This file will take user integer input (inValue), and ~~the~~ reverse ~~of~~ it
//

#include <iostream>
using namespace std;

~~// Declare and set RADIX~~ (base number of decimal system)
const int RADIX = 10;

int main()
{
   ~~// Declare User Input (inValue), most significant digit (msd)~~
   int inValue, dividend, divisor = 1;

   // Request and Store User Input
   cout << "Enter any decimal integer value: ";
   cin >> inValue;

   // Display Opening Phrase
   cout << "\"" << inValue << "\" in words is  \"";

   // Force to be positive AND display "minus"
   if (inValue < 0)
   {
      inValue = -inValue;
      cout << "minus";
   }

   dividend = inValue;

   // Find the power of ten of the input
   do
   {
      divisor *= RADIX;
      dividend /= RADIX;
   } while (dividend >= RADIX);

   dividend = inValue;

   do
   {
      // Determine the most significant digit
      int msd = dividend / divisor;

      // Write out most significant digit in words
      switch (msd)
      {
         case 0:
            cout << " zero";
            break;
         case 1:
```

Please don't comment the obvious. It just needlessly clutters your code.

?

```
            cout << " one";
            break;
         case 2:
            cout << " two";
            break;
         case 3:
            cout << " three";
            break;
         case 4:
            cout << " four";
            break;
         case 5:
            cout << " five";
            break;
         case 6:
            cout << " six";
            break;
         case 7:
            cout << " seven";
            break;
         case 8:
            cout << " eight";
            break;
         case 9:
            cout << " nine";
            break;
         default:
            cout << " UNRECOGNIZED DIGIT";
      }

      // reduce the most significant digit
      dividend -= msd * divisor;

      // Remove the LSB
      divisor /= RADIX;
   } while (divisor != 0);

   cout << "\"\n";
   return 0;
}
```

```
********** C1 ASSIGNMENT 3 EXERCISE 3 AUTOMATIC PROGRAM RUN RESULTS **********

************* THE RESULTS BELOW HAVE BEEN PARTIALLY CHECKED AND **************
*************   NO ERRORS WERE FOUND.  HOWEVER, THIS DOES NOT   **************
*************  NECESSARILY MEAN THAT THERE ARE NO ERRORS.  THE  **************
*************   THE INSTRUCTOR WILL DO A MORE THOROUGH CHECK    **************
*************              DURING MANUAL GRADING.               **************


------------------------------ START OF 1ST RUN ------------------------------

Enter any decimal integer value: 3987
"3987" in words is  " three nine eight seven"

------------------------------- END OF 1ST RUN -------------------------------

------------------------------ START OF 2ND RUN ------------------------------

Enter any decimal integer value: -2645
"-2645" in words is  "minus two six four five"

------------------------------- END OF 2ND RUN -------------------------------

------------------------------ START OF 3RD RUN ------------------------------

Enter any decimal integer value: 100
"100" in words is  " one zero zero"

------------------------------- END OF 3RD RUN -------------------------------

------------------------------ START OF 4TH RUN ------------------------------

Enter any decimal integer value: 000120
"120" in words is  " one two zero"

------------------------------- END OF 4TH RUN -------------------------------

------------------------------ START OF 5TH RUN ------------------------------

Enter any decimal integer value: -0023
"-23" in words is  "minus two three"

------------------------------- END OF 5TH RUN -------------------------------

------------------------------ START OF 6TH RUN ------------------------------

Enter any decimal integer value: 000
"0" in words is  " zero zero"

------------------------------- END OF 6TH RUN -------------------------------

------------------------------ START OF 7TH RUN ------------------------------

Enter any decimal integer value: 1010
"1010" in words is  " one zero one zero"

------------------------------- END OF 7TH RUN -------------------------------

------------------------------ START OF 8TH RUN ------------------------------
```

```
Enter any decimal integer value: -1010
"-1010" in words is  "minus one zero one zero"

------------------------------- END OF 8TH RUN -------------------------------

------------------------------ START OF 9TH RUN ------------------------------

Enter any decimal integer value: 3
"3" in words is  " zero three"

------------------------------- END OF 9TH RUN -------------------------------
```