Tutorial Letter 103/1/2018

Introduction to Programming I

COS1511

Semester 1

School of Computing

This tutorial letter Assignment 3 for Semester 1.

BARCODE

CONTENTS

Page

1 INTRODUCTION

Dear student,

Welcome to COS1511. We hope that you will find it interesting and stimulating. There are a number of first-year Computer Science and Information Systems modules at Unisa. Here are some of the modules.

Because this is a blended online module, you need to go online to see your study materials and read what to do for the module. Go to the *myUnisa* website here: https://my.unisa.ac.za and login with your student number and password. You will see **COS1511-18-S1** (for the first semester) or **COS1511-18-S2** (for the second semester) in the row of modules in the orange blocks across the top of the webpage. Remember to also check in the -**more**- tab if you cannot find it in the orange blocks. Click on the module you want to open.

In addition, you will receive this tutorial letter and a printed copy of the online study materials from your module. While these printed materials may appear to be different from the online study materials, they are exactly the same and have been copied from the online *myUnisa* website.

2 Calculation of the semester mark

The marks that you obtain for Assignments 1, 2 and 3 form the semester mark for COS1511. The semester mark forms 20% of the final mark for the module. The weights of the COS1511 assignments are indicated in the table below:

Assignment number	Weight
1	30%
2	40%
3	30%

An example follows: Suppose a student gets 60% for Assignment 1, 45% for Assignment 2 and 65 for Assignment 3. In order to calculate the semester mark, the mark obtained for the specific assignment is multiplied by the weight. This then forms part of the 20% that the semester mark contributes to the final mark. Therefore:

Assignment	Marks obtained	Weight	Contribution to semester mark	
1	60%	30%	60/100 x 30/100 x 20	3.6
2	68%	40%	68/100 x 40/100 x 20	5.4
3	65%	30%	65/100 x 30/100 x 20	3.0
TOTAL				12.0

In this example the student has a semester mark of 12.0 out of 20. **The semester mark will not form part of the final mark of a supplementary examination.**

3 Due dates of assignment

The table below gives the due dates of the assignments for this module.

Assignment	**Due Date 1st semester**	**Weight**
~~1~~	~~19 March~~	~~30%~~
~~2~~	~~9 April~~	~~40%~~

3	26 April	30%

3.1 Submission of assignments

Students must submit assignments 1 and 3 (completed on a mark-reading sheet) **either** by Mobile MCQ submission on your cell phone **or** electronically via *myUnisa*. For detailed information and requirements as far as assignments are concerned, see *Studies@Unisa*, which you received with your study package. Follow the instructions given in Tutorial Letter COSALLF/301/4/2018, as well as the brochure *Studies@Unisa*, when submitting your assignments. The URL for *myUnisa* is: http://my.unisa.ac.za/. Instructions on how to register to become a *myUnisa* user, and how you should format your assignments before you submit them electronically, are given on the website. The two most important things to remember are that your submission must consist of a single text file, and that you may submit an assignment only once.

To submit an assignment through *myUnisa*:

- go to *myUnisa*
- log in with your student number and password
- select the module
- click on assignments in the menu on the left-hand side of the screen
- click on the assignment number that you wish to submit
- follow the instructions

PLEASE NOTE: Assignments can be tracked (e.g. whether or not the University has received your assignment or the date on which an assignment was returned to you) on *myUnisa*.

4 Assignment 3

Assignments 3 consist of multiple choice questions (MCQs). The assignments have to be submitted ON TIME. The assignments can either submitted by Mobile MCQ submission or electronically via *myUnisa*.

SUBMISSION: **Either** by Mobile MCQ submission **or** electronically via *myUnisa*.

Please note that Assignment 1 has to be submitted ON TIME in order to gain examination admission. It will be to your own advantage to check after a few days whether the assignment has been registered on the system. If you have not completed the assignment by the due date, submit whatever you have completed – you will get marks for everything that you have done.

If you want to submit the assignment electronically and *myUnisa* is off-line during that time, you need not contact us, because we will be aware of it. Simply submit it as soon as *myUnisa* is available again.
Note the assignment unique number below – if you submit through *myUnisa* you will be asked to enter it. If you submit by post, please fill it in where it is requested on the mark reading sheet.

MULTIPLE CHOICE ASSIGNMENT

SUBMISSION: **Either** by Mobile MCQ submission **or** electronically via *myUnisa*.
Please note that Assignment 1 has to be submitted ON TIME in order to gain examination admission. It will be to your own advantage to check after a few days whether the assignment has been registered on the system. If you have not completed the assignment by the due date, submit whatever you have completed – you will get marks for everything that you have done.

If you want to submit the assignment electronically and *myUnisa* is off-line during that time, you need not contact us, because we will be aware of it. Simply submit it as soon as *myUnisa* is available again.

Question 1

With arrays, the data type of the index is always a(n) _____.

1. `*int`
2. `float`
3. `char`
4. `string`

Question 2

The first variable in a one-dimensional array is assigned an index of _____.

1. 0
2. 1
3. either a or b
4. none of the above

Question 3

Which of the following statements declares a one-dimensional array containing two integer values?

1. `int numbers[3] = {1, 2, 3};`
2. `char letters[2] = {'a', 'b'};`
3. `int characters[2] = {45, 87};`
4. `int ages[6] = {0};`

Question 4

Which of the following statements will store the user's entry into the second element of the `numbers` array?

1. `numbers[1] = 0;`
2. `numbers[2] = 0;`
3. `cin >> numbers[1];`
4. `cin >> numbers[2];`

Question 5

Arrays in C++ are passed automatically _____ to functions.

1. by value
2. by reference
3. by actual parameters
4. by forma parameters

Question 6

When passing an array to a function, you code _____ after the formal parameter name of the array in the function header to indicate it is an array.

1. `&`
2. `[]`
3. `{}`
4. `||`

Question 7

Passing an array _____ allows the computer to pass the address of only the first array element.

1. by value
2. by reference
3. by address
4. by scalar

Question 8

Which of the following statements declares a one-dimensional array called `letters` to contain the first five lowercase letters of the alphabet.

```
1. char letters[5] = {'a', 'b', 'c', 'd', 'e'};
2. char letters[5] = {"a", "b", "c", "d", "e"};
3. char letters[5] = {'a'; 'b'; 'c'; 'd'; 'e'};
4. char letters[5] = {"a"; "b"; "c"; "d"; "e"};
```

Question 9

What happens when an array declaration statement does not provide an initial value for each of the elements in a numeric array?

1. The compiler automatically initialises the elements, so the array elements contain 0.
2. The compiler does not automatically initialise the elements, but as it is a numeric array, all array elements contain a numeric value.
3. The compiler does not automatically initialise the elements, so the array elements may contain garbage.
4. The compiler automatically initialises the elements with 0.

Question 10

Given the array declaration below, what is stored in numbers[2] ?

```
int numbers[5] = {1, 2, 3, 4, 5};
```

```
1. 0
2. 1
3. 2
4. 3
```

Question 11

Given the array declaration below, which assignment statement will place the number 5 into the fifth array element?

```
int numbers[5] = {0};
```

```
1. numbers[4] = 5;
2. numbers[5] = 5;
3. numbers[1+3] = 5;
4. numbers[1+4] = 5;
```

Question 12

Given the array declaration below and user prompt, which statement will place the number 1 into the first array element using the extraction operator.

```
int numbers[5] = {0};
cout << "Enter the first number: ";
```

```
1. cin >> numbers[1];
2. cin >> numbers[0] + 1;
3. cin >> numbers[0] = 1;
4. cin >> numbers[0];
```

Question 13

Variables located in the first column in a two-dimensional array are assigned a column subscript of _____.

1. 0
2. 1
3. -1
4. NULL

Question 14

Variables located in the third row in a two-dimensional array are assigned a row subscript of _____.

```
1. 2
2. 3
3. 4
4. -3
```

Question 15

How many elements will a two-dimensional array with 4 rows and 5 columns contain?

```
1. 9
2. 12
3. 18
4. 20
```

Question 16

Which of the following statements will declare a two-dimensional array called `scores` to be of the `int` data type and contain 3 rows and 4 columns, while also initializing all elements to 0?

```
1. int scores [4][3] = 0;
2. int scores [3][4] = 0;
3. int scores[3][4] = {0};
4. int scores[12] = {0};
```

Question 17

When referencing an element in a two-dimensional array, the _____ is coded before the _____.

```
1.  column index, row index
2.  row index, column index
3.  column index, array name
4.  row index, array name
```

Question 18

How many loops are necessary to access the contents of a two-dimensional array efficiently?

1. 0
2. 1
3. 2
4. more than 2

Question 19

How many loops are necessary to access the contents of a one-dimensional array efficiently?

1. 0
2. 1
3. 2
4. more than 2

Question 20

Given the following array declaration, what is the value stored in the `scores[2][3]` element?

```
int scores[5][5] = {5};
```

1. 0
2. 5
3. 10
4. 25

Question 21

Given the following array declaration, what is the value stored in the scores[1][1] element?

```
int scores[5][5] = {5};
```

1. 0
2. 5
3. 10
4. 25

Question 22

Given the following array declaration, what is the value stored in the `scores[1][1]` element?

```
int scores[3][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };
```

1. 1
2. 4
3. 7
4. none of the above

Question 23

Given the following array declaration, what is the value stored in the `scores[2][2]` element?

```
int scores[3][3] = { {1, 2, 3} };
```

1. 0
2. 1
3. 2
4. 3

Question 24

You would use _____ to reference the variable located in the first row, first column of the `scores` two-dimensional array.

1. `scores[0][0]`
2. `scores[1][1]`
3. `scores[first][first]`
4. `scores[1,1]`

Question 25

Which statement will create a two-dimensional array called `orders` of type `int` with 2 rows and 3 columns and intialises each element to 0?

1. `int orders[2][3];`
2. `int orders[][] = {0};`
3. `int orders[3][2] = {0};`
4. `int orders[2][3] = {0};`

Question 26

Which statement will create a 2x2 char array called `letterswith` and initialses each element '`x`'?

1. `char letters[][2] = { {'x', 'x'}, {'x', 'x'} };`
2. `char letters[2][2] = { {'x', 'x'}, {'x', 'x'} };`
3. `char letters[2][] = { {'x', 'x'}, {'x', 'x'} };`
4. `char letters[2][2] = {'x'};`

Question 27

Which statement will create a two-dimensional integer array called `scores` with 3 rows and 2 columns where each element has the initial value of 100?

1. `int scores[3][2] =  { {100, 100}, {100, 100}, {100, 100} };`
2. `int scores[][] =  {100};`
3. `int scores[2][3] =  { {100, 100}, {100, 100}, {100, 100} };`
4. `int scores[3][2] =  {100};`

Question 28

Given the declaration below, which statement assigns the element in the first row and first column a value of 99?

```
int scores[3][2] = {0};
```

```
1. scores[1][1] = 99;
2. scores[0][0] = '99';
3. scores[0][0] = 99;
4. scores[1][1] = "99";
```

Question 29

Given the declaration below, which statement assigns the element in the last row and last column a value of 99?

```
int scores[6][7] = {0};
```

```
1. scores[4][5] = 99;
2. scores[6][7] = 99;
3. scores[5][6] = 99;
4. scores[5][6] = '99';
```

Question 30

Given the declaration below, which statement assigns the element in the first row and first column the value entered by the user through the prompt below.

```
int scores[3][2] = {0};
cout << "Please enter the first score: ";
```

```
1. cin >> scores[1][1];
2. cin >> scores[0][0];
3. cin >> scores[1][0];
4. cin >> scores[0][1];
```

Question 31

Given the declarations below, which nested loop sums all of the elements in the array below.

```
int scores[3][3] = { {92, 87, 91}, {88, 72, 93}, {100, 94, 97} };
int sum = 0;
```

```
1. for (int row = 0; row <= 2; row = row + 1)
     for (int col = 1; col <= 3; col = col + 1)
          sum = sum + scores[row][col];

2. for (int row = 0; row <= 2; row = row + 1)
     for (int col = 0; col <= 2; col = col + 1)
          sum = sum + scores[row][col];

3. for (int row = 0; row <= 2; row = row + 1)
     for (int col = 0; col <= 2; col = col + 1)
          sum = sum + scores[col][row];

4. for (int row = 1; row <= 3; row = row + 1)
     for (int col = 0; col <= 2; col = col + 1)
          sum = sum + scores[row][col];
```

Question 32

What is the output of the program below?

```
#include<iostream.h>
void main()
{
     int n=1;
     cout<<endl<<"The numbers are;"<<endl;
     do
     {
             cout <<n<<"\t";
             n++;
         } while (n<=100);
     cout <<endl;
 }
```

1. print natural numbers 0 to 99
2. print natural numbers 1 to 99
3. print natural numbers 0 to 100
4. print natural numbers 1 to 100

Question 33

..... allows that a section of a program is compiled only if the defined constant that is specified as the parameter has been defined, independently of its value.

1. `#ifdef`
2. `#if`
3. `#define`
4. `#ifd`

Question 34

____ is not one of the fundamental data types in C++.

1. `int`
2. `char`
3. `float`
4. `string`

Question 35

The statement ____ declares and initializes a `string` variable named `zipCode`

1. `string zipCode = "";`
2. `zipCode = "";`
3. `init zipCode = "";`
4. `declare zipCode = "";`

Question 36

A call to the `getline` function contains a maximum of _____ parameters.

1. 0
2. 1
3. 2
4. 3

Question 37

A call to the `substr` function contains a maximum of _____ parameters.

1. 0
2. 1
3. 2
4. 3

Question 38

A call to the `find` function contains a maximum of _____ parameters.

1. 0
2. 1
3. 2
4. 3

Question 39

A call to the `erase` function contains a maximum of _____ parameters.

1. 0
2. 1
3. 2
4. 3

Question 40

You can use the string class's _____ function to remove one or more characters located anywhere in a string variable.

1. `remove`
2. `delete`
3. `truncate`
4. none of the above

Question 41

You can use the string class's _____ function for inserting characters within a string variable.

1. `add`
2. `addition`
3. `append`
4. `insert`

Question 42

Given the following, which statement stores the characters entered by the user, up until the newline character, in the `name` variable; and consume the newline character.

```
string name = "";
cout << "Enter your name: ";
```

1. `getline(cin, name);`
2. `getline( name, cin);`
3. `readline(cin, name);`
4. `cin >> (cin, name);`

Question 43

Which C++ statements determine the length of the `name` string defined below and place the value into the `number` variable?

```
int number = 0;
string name = "Hello";
```

1. `number = name.len();`
2. `number = name.size();`
3. `number = size(name);`
4. `number == name.size();`

Question 44

Given the declarations below, using the `substr` function, which statements will assign "`First`" to the `name1` variable and "`Last`" to the `name2` variable.

```
string name = "First Last"; string name1 = "";
string name2 = "";
```

1.
```
name1 = name.substr(1, 5);
name2 = name.substr(6);
```

2.
```
name1 = name.substr(1, 6);
name2 = name.substr(6);
```

3.
```
name1 = name.substr(0, 5);
name2 = name.substr(6);
```

4.
```
name1 = name.substr(0, 5);
name2 = name.substr(5);
```

Question 45

Given the declarations below, using the `find` function, write the statements to search the `number` string variable to determine if the sequence "`123`" exists. The location should be placed in the `position` variable.

```
int position = 0;
string number = "1234567890";
```

```
1. position = number.find(123);
2. find("123", 0);
3. position = number.find("123", 0);
4. find(number("123", 0), position);
```

Question 46

Which one of the following options represents the output of the program below?

```
struct pixel
{
     int c, r;
};

void display(pixel p)
{
     cout << "col "<< p.c << " row " << p.r << endl;
}

int main()
{
     pixel x = {40,50}, y, z;
     z = x;
     x.c += 10;
     y = z;
     y.c += 10;
     y.r += 20;
     z.c -= 15;
     display(x);
     display(y);
     display(z);
     return 0;
}
```

```
1. Col 50 Row 50
   Col 50 Row 70
   Col 25 Row 50

2. Col 40 Row 40
   Col 40 Row 70
   Col 25 Row 50

3. Col 60 Row 50
   Col 60 Row 80
   Col 25 Row 50

4. Col 50 Row 70
   Col 50 Row 55
   Col 25 Row 50
```

Question 47

Which one of the following options represents the output of the program below?

```
struct play
{
     int score, bonus;
};

void calculate(play &p, int n = 10)
{
     p.score++;
     p.bonus += n;
}

int main()
{
     play pl = {10, 15};
     calculate(pl, 5);
     cout << pl.score << ":" << pl.bonus << endl;
     calculate(pl);
     cout << pl.score << ":" << pl.bonus << endl;
     calculate(pl, 15);
     cout << pl.score << ":" << pl.bonus << endl;
     return 0;
}
```

1. 11:20
   12:30
   13:45

2. 16:10
   17:40
   18:55

3. 10:20
   11:30
   12:45

4. 11:10
   12:20
   13:35

Question 48

Which one of the following options represents the output of the program below?

```
struct myBox
{
      int length, breadth, height;
};

void dimension (myBox M)
{
     cout << M.length << "x" << M.breadth << "x";
     cout << M.height << endl;
}

     int main ()
     {
          myBox B1 = {10, 15, 5}, B2, B3;
          ++B1.height;
          dimension(B1);
          B3 = B1;
          ++B3.length;
          B3.breadth++;
          dimension(B3);
          B2 = B3;
          B2.height += 5;
          B2.length--;
          dimension(B2);
          return 0;
     }
```

1. 10x15x5
   11x16x6
   10x16x11

2. 10x5x6
   11x16x6
   10x16x11

3. 10x15x6
   11x16x6
   10x5x11

4. 10x15x6
   11x16x6
   10x16x11

Question 49

What is the output produced by the program below?

```
#include <iostream>
using namespace std;

void encrypt(char theChar[])
{
     for (int i = 0; theChar[i] != '\0'; i += 2)
          if (theChar[i] == 'A' || theChar[i] == 'E')
               theChar[i] = '#';
          else if (islower(theChar[i]))
               theChar[i] = toupper(theChar[i]);
          else
               theChar[i] = '@';
}
int main()
{
     char text[]="SaVE EArtH";
     encrypt(text);
     cout << text << endl;

     return 0;
}
```

1. #a#E#E@rTH
2. @a@E@E#rtH
3. @a@E@E#rTH
4. #a#E#E#rtH

Question 50

What is the output produced by the program below?

```
#include <iostream>
#include <string>
using namespace std;

int main()
{
    string name = "CoMPutER";
    for (int x = 0; x < name.size(); x++)
        if (islower(name [x]))
            name [x] = toupper(name[x]);
        else
            if (isupper(name[x]))
                if (x % 2 == 0)
                    name[x] = tolower(name[x]);
                else
                    name[x] = name[x-1];
     cout << name;

     return 0;
}
```

1. CoMPutEr
2. cOmpUTeR
3. coMMutEE
4. cOmmUTee

Question 51

What is the output of the following program?

```
#include <iostream>
using namespace std;

void X(int A, int &B)
{
     A = A + B;
     B = A - B;
     A = A - B;
}

int main()
{
     int a = 4, b = 18;
     X(a,b);
     cout << a << ", " << b;

     return 0;
}
```

1. 4, 4
2. 18, 4
3. 18, 18
4. 4, 18

Question 52

What is the output of the following program?

```
#include <iostream>
using namespace std;

void func(int &x, int y)
{
     x = x - y;
     y = x * 10;
}

int main()
{
     int value = 7;
     func (value, value);
     cout << value << ", " << value << '\n';

     return 0;
}
```

1. 7, 70
2. 7, 0
3. 0, 0
4. 0, 70

Question 53

What is the output of the following program?

```
#include <iostream>
using namespace std;

int i = 100;
void abc()
{
     int i = 8;
     cout << "first = " << i++ << endl;
}

int main()
{
     int i = 2;
     abc();
     cout << "second = " << i << endl;
     abc();

     return 0;
}
```

1. first = 8
   second = 2
   first = 8

2. first = 8
   second = 2

3. second = 2
   first = 8

4. first = 8
   second = 2
   first = 9

Question 54

What is output of the following program?

```
#include <iostream>
using namespace std;

int func(int &x, int y = 10)
{
     if (x % y == 0)
          return ++x;
     else
          return y--;
}
```

```
int main()
{
     int p = 20, q = 23;

     q = func(p, q);
     cout << p << " " << " " << q << endl;

     p = func (q);
     cout << p << " " << " " << q << endl;

     q = func (p);
     cout << p << " " << " " << q << endl;

     return 0;
}
```

1.
```
20 22
10 23
11 11
```

2.
```
20 23
10 23
11 11
```

3.
```
20 23
9 23
11 11
```

4.
```
20 23
10 23
10 11
```

Question 55

Which C++ program segment will display the following pattern?

```
**********
**********
**********
**********
```

1.
```
int i, j;
for(i = 1; i <= 4; i++)
{
   for(j = 1; j <= 10; j++)
        cout << '*' << endl;
}
```

2.
```
int i, j;
for(i = 1; i < 4; i++)
{
   for(j = 1; j <= 10; j++)
          cout << '*';
   cout << endl;
}
```

3.
```
int i, j;
for(i = 1; i <= 4; i++)
{
   for(j = 1; j <= 10; j++)
       cout << '*';
   cout << endl;
}
```

4.
```
int i, j;
for(i = 0; i <= 4; i++)
{
   for(j = 1; j < 10; j++)
         cout << '*';
   cout << endl;
}
```

The following four questions refer to the `person` structure.

Question 56

Which of the following declarations define a structure named `person` which has three members: `name`, `age` and `salary`.

```
1.   struct
     {
        string name;
        int age;
        float salary;
     } person;

2.   struct person
        string name;
        int age;
        float salary;

3. struct person
     {
        string name;
        int age;
        float salary;
     };

4.   person
     {
        string name;
        int age;
        float salary;
     };
```

Question 57

Which of the following statements define a variable of type `person`?

```
1. person bill;
2. bill person;
3. person bill[];
4. person[] bill;
```

Question 58

Which statement accesses the `age` member of structure variable `bill` and assign the value 50 to it.

```
1. bill[].age = 50;
2. bill[age] = 50;
3. bill(age) = 50;
4. bill.age = 50;
```

Question 59

Which one of the following statements declares an array of size 10 with elements of type `person`?

```
1. person bill[10];
2. person[] bill;
3. person bill[9];
4. person[10] bill;
```

Question 60

What is output of the following program?

```
#include <iostream>

using namespace std;
int var = 10;

int func()
{
     return var;
}

int main()
{
     int var = 5;
     cout << func()  << "\n";
     return 0;
}
```

1. 0
2. 15
3. 10
4. 5

Question 61

Which of the following function header lines is valid for a function called `findMax` that finds and returns the maximum value stored in an array of integers that is passed in as a parameter?

a. `int findMax(int [])`
b. `int findMax(int values[])`
c. `int findMax(int)`
d. `int findMax([])`

Question 62

Assuming the following declarations:

```
const int NUMROWS = 3;
const int NUMCOLS = 4;
int val[NUMROWS][NUMCOLS] = {8,16,9,52,3,15,27,6,14,25,2,10};
```

Which of the following loops correctly outputs each element of the array in row order?

1.
```
for (i = 0; i < NUMROWS; i++)
   {
   for (j = 0; j < NUMCOLS; j++)
      cout << setw(4) << val[i][j];
      cout << endl;
   }
```

2.
```
for (i = 0; i < NUMROWS; i++)
   {
      cout << setw(4) << val[i][j];
      cout << endl;
   }
```

```
3. for (i = 0; i < NUMCOLS; i++)
   {
      cout << setw(4) << val[i][j];
      cout << endl;
   }

4. for (i = 0; i < NUMROWS*NUMCOLS; i++)
   {
      cout << setw(4) << val[i][j];
      cout << endl;
   }
```

Question 63

Which of the following function header correctly includes a two dimensional array as its parameter?

```
1. void display(int [][COLS]);
2. void display(int [ROWS][]);
3. void display(int [][]);
4. void display(int [ROWS][COLS]);
```

Question 64

What is the output of this program?

```
#include <iostream>
using namespace std;
int main ()
{
     int array[] = {0, 2, 4, 6, 7, 5, 3};
     int n, result = 0;
     for (n = 0; n < 8; n++)
     {
          result += array[n];
     }
     cout << result;
     return 0;
}
```

1. 24
2. 25
3. 26
4. 27

Question 65

What is the output of this program?

```
#include <iostream>
using namespace std;
int main()
{
     char str[5] = "ABC";
     cout << str[3];
     cout << str;
     return 0;
}
```

1. ABC
2. ABCD
3. AB
4. C

Question 66

Suppose that `gamma` is an array of 50 elements of type `int` and `j` is an `int` variable. Which of the following `for` loops sets the subscript of `gamma` out-of-range?

```
1. for (j = 0; j <= 49; j++)
      cout << gamma[j] << " ";
2. for (j = 1; j < 50; j++)
      cout << gamma[j] << " ";
3. for (j = 0; j <= 50; j++)
      cout << gamma[j] << " ";
4. for (j = 0; j <= 48; j++)
      cout << gamma[j] << " ";
```

Question 67

Which of the following `struct` definitions is correct in C++?

```
1. struct studentType
   {
      int ID;
   };

2. struct studentType
   {
      string name;
      int ID;
      float gpa;
   }

3. int struct studentType
   {
      ID;
   }

4. struct studentType
   {
      int ID = 1;
   };
```

Question 68

Consider the following `struct` definition:

```
struct rectangleData
{
    float length;
    float width;
    float area;
    float perimeter;
};
```

Which of the following variable declarations is correct?

```
1. rectangle rectangleData;
2. struct rectangleData();
3. rectangleData myRectangle;
4. rectangleData rectangle = new rectangleData();
```

Question 69

Consider the following `struct` definition:

```
struct temp
{
     int b;
};
temp s[50];
```

The correct syntax to access the member of the `i`th structure in the array of structures is?

```
1. s.b.[i];
2. s.[i].b;
3. s.b[i];
4. s[i].b;
```

Question 70

The syntax for accessing a `struct` member is `structVariableName ____`.

```
1. .memberName
2. *memberName
3. [memberName]
4. $memberName
```

Question 71

Consider the following statements:

```
struct rectangleData
{
    float length;
    float width;
    float area;
    float perimeter;
};
rectangleData bigRect;
```

Which of the following statements correctly initializes the element `length` of `bigRect`?

```
1. bigRect = {10};
2. bigRect.length = 10;
3. length[0]= 10;
4. bigRect[0]= 10
```

Question 72

Consider the following statements:

```
struct rectangleData
{
    float length;
    float width;
    float area;
    float perimeter;
};
rectangleData bigRect;
```

Which of the following statements is valid in C++?

```
1. cin >> bigRect;
2. cin >> bigRect.length;
3. perimeter = 2 * (length + width);
4. area = length * width;
```

Question 73

Consider the following statements:

```
struct personalInfo
{
    string name;
    int age;
    float height;
    float weight;
};

struct commonInfo
{
    string name;
    int age;
};
personalInfo person1, person2;
commonInfo person3, person4;
```

Which of the following statements is valid in C++?

```
1. person1 = person3;
2. person2 = person1;
3. person2 = person3;
4. person2 = person4;
```

Question 74

Consider the following statements:

```
struct studentType1
{
    string name;
    int ID;
    float gpa;
};

struct studentType2
{
    string name;
    int ID;
    float gpa;
};
studentType1 student1, student2;
studentType2 student3, student4;
```

Which of the following statements is valid in C++?

```
1. student2 = student3;
2. student1 = student4;
3. student2.ID = ID;
4. student1.ID = student3.ID;
```

Question 75

Consider the following statements:

```
struct rectangleData
{
    float length;
    float width;
    float area;
    float perimeter;
};

rectangleData bigRect;
rectangleData smallRect;
```

Which of the following statements is legal in C++?

```
1. if (bigRect == smallRect)
2. if (bigRect != smallRect)
3. if (bigRect.length == width)
4. if (bigRect.length == smallRect.width)
```

Question 76

Consider the following statements:

```
struct circleData
{
   float radius;
   float area;
   float circumference;
};
circleData circle;
```

Which of the following statements is valid in C++?

```
1. cin >> circle.radius;
   circle.area = 3.14 * radius * radius;
2. cin >> circle.radius;
   circle.area = 3.14 * circle.radius * radius;
3. cin >> circle;
4. cin >> circle.radius;
```

Question 77

Consider the following statements:

```
struct rectangleData
{
    float length;
    float width;
    float area;
    float perimeter;
};
rectangleData bigRect;
```

Which of the following statements is valid in C++?

```
1. cin >> bigRect.length >> width;
2. cout << bigRect.length;
3. cout << bigRect;
4. cout << length;
```

Question 78

Consider the following statements:

```
struct supplierType
{
    string name;
    int supplierID;
};

struct applianceType
{
    supplierType supplier;
    string modelNo;
    float cost;
};
applianceType applianceList[25];
```

Which of the following best describes `applianceList`?

1. It is a multidimensional array.
2. It is a `struct`.
3. It is an array of `structs`.
4. It is a `struct` of arrays.

Question 79

Consider the following statements:

```
struct supplierType
{
    string name;
    int supplierID;
};

struct applianceType
{
    supplierType supplier;
    string modelNo;
    float cost;
};

applianceType applianceList[25];
```

Which of the following statements correctly initializes the cost of each appliance to `0`?

```
1. applianceList.cost = 0;
2. applianceList.cost[25] = 0;
3. for (int j = 1; j < 25; j++)
            applianceList.cost[j] = 0;

4. for (int j = 0; j < 25; j++)
            applianceList.cost[j] = 0;
```

Question 80

What is the output of this program?

```
#include <iostream>
#include <string>
using namespace std;

struct student
{
     int num;
     char name[25];
};

int main()
{
     student stu;
     stu.num = 123;
     strcpy(stu.name, "John");
     cout << stu.num << endl;
     cout << stu.name << endl;
     return 0;
}
```

1. 123
   John
2. John
   John
3. 123
   123
4. compile time error

End of Assignment 3