National University of Computer and Emerging Sciences

School of Computing **Fall 2013** **Islamabad Campus**

CS101
Introduction to Computing

Monday, December 16, 2013

Course Instructor

Dr. Shahzad Rajput and Dr. Sibt ul Hussain

Serial No:

Final Exam
Total Time: 3 Hour
Total Marks: 1000

_____
Signature of Invigilator

_____ _____ _____ _____

Student Name Roll No Section Signature

DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.

Instructions:

1. Attempt on question paper. Attempt all of them. Read the question carefully, understand the question, and then attempt it.
2. No additional sheet will be provided for rough work. Use the back of the last page for rough work.
3. If you need more space write on the back side of the paper and clearly mark each question and part number etc.
4. After asked to commence the exam, please verify that you have **(18)** different printed pages including this title page. There are total of **eight (8)** questions.
5. Use of **calculator** is strictly **prohibited.**
6. Draw a smiley at the bottom of this page and earn ten bonus marks.
7. Use permanent ink pens only. Any part done using **soft pencil** will not be marked and cannot be claimed for rechecking.
8. Use **proper indentation and comments** while writing code and make sure that your code is legible. Failing to do so can cost you marks.

	Q-1	Q-2	Q-3	Q-4	Q-5	Q-6	Q-7	Q-8	**Total**
Marks Obtained									
Total Marks	60	220	320	100	50	50	100	100	**1000**

Vetted By: _____Vetter Signature: _____

Q. No. 1 [6*10=60 marks]

(a) What is the binary representation of 355_{10}?

(b) What is the hexadecimal representation of 1033_8?

(c) Given that 125_{10}= 0111 1101_2, what is the signed binary representation of -120_{10}

(d) Given that $3072_{10} = 6*2^9$ and $48_{10} = 3*2^4$, what is the binary representation of 3130_{10}?

(e) What is the 16-bit floating point representation of 10101.0101110110111_2 in normalized binary form using the following convention?

Sign of mantissa = left most bit (where 0: +; 1: -)
Mantissa = next 11 bits, leading 1 is hidden, really represents 12 bits
Exponent = next four bits, bias 7

(f) Interpret the following number given in floating point representation

$0101\ 1110\ 0000\ 1001_2$

using the convention mentioned below:

Sign of mantissa = left most bit (where 0: +; 1: -)
Mantissa = next 11 bits, leading 1 is hidden, really represents 12 bits
Exponent = next four bits, bias 7

Find its decimal equivalent.

Answer:

Q. No. 2 [11*20=220 marks]

Write the output of the given C++ code segments.

(a) Output

```
int sum(int i, int j){
        return (i+j+5/2);
}

int main(){
        int i = 7.5, j=4.5;
        cout << sum(i,j) << endl;
        return 0;
}
```

(b) Output

```
for( int i=0; i<7; i++ )
        cout << i%3 << " ";
        cout << endl;
```

(c) Output

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

int main(){
        number = 62.7;
        double number;
        cout << number << endl;
        return 0;
}
```

(d) Output

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

int main(){
        int tV, fV;
        int x = 5, y = 10;

        tV = x < y;
        fV = y == x;

        cout << tV << endl << fV << endl;
        return 0;
}
```

(e) Output

```
int num=10;
if( num ){
        int num = 20;
        num *= 100;
        num += 1000;
}
cout << num << endl;
```

(f) Output

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

int x = 5;

void test(){
        cout << ++x;
}

int main(){
        int x = 1000;
        for( int i = 1; i<5; i++ )
                test();
                cout << x << endl;
        return 0;
}
```

(g) Output

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

int magic( int x=10, int y=20 ){
        if( x == 10 )
                return( y );
        return( x + y );
}

int main(){
        cout    << magic() << endl
                << magic(100) << endl
                << magic(1000, 2000) << endl;
        return 0;
}
```

(h) Output

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

int magic( int &x, int y ){
        x = 2000;
        y = 500;
        return (x-y);
}

int main(){
        int i=500, j=600, k;
        k = magic( i, j );
        cout    << i << " " << j << " " << k << endl;

        return 0;
}
```

(i) Output

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

```
void magic( int *p, int s ){
        int i = 0;
        while( s-- > 0 ){
                i += *(p++);
        }
        cout << i << endl;
}

int main(){
        int arr[] = { 1, 2, 3, 4, 5 };
        magic( arr, 5 );
        return 0;
}
```

(j) Output

```
int arr[] = { 10, 20, 30, 40, 50, 60, 70 };
int *p = &arr[1];
p = p + 4;
cout << *p << endl;
```

(k) Output

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

int main(){
        const int M = 10;
        int N = 2*M;
        int SH = 10;
        int A = 0;

        int p[M];

        for( int i=0; i<M; i++ )
                p[ i ] = i*10-5;

        for( int i=1; i<=N; i++ )
                *(p+ (i%M) ) += SH;

        for( int i=0; i<M; i++ )
                A += p[ i ];

        cout << A << endl;

        return 0;
}
```

Q. No. 3 [320 marks]

a) [20 marks]

Consider you have two variables: `sales` and `commissionRate`. Write an `if` statement that performs the following logic: if the variable `sales` is outside the range [50,000 – 100,000] then assign 0.25 to the `commissionRate` variable, and assign 250 to the `bonus` variable.

b) [20 marks]

Write a `for` loop that displays all of the odd numbers, 1 through 49.

c) [20 marks]

Convert the following `for` loop to a `while` loop:

```
for (int x = 50; x > 0; x--){
        cout << x << " seconds to go.\n";
}
```

d) [20 marks]

Write a function named `tenTimes`. The function should have an integer parameter named `number`. When `tenTimes` is called, it should return the tenth power of the number (i.e., $number^{10}$).

e) [40 marks]

Write a nested `for` loop that displays the following output on screen:

```
6   1   6
7   2   14
8   3   24
6   1   6
7   2   14
6   1   6
```

f) [20 marks]

Complete the following program skeleton by writing a `switch` statement that displays "one" if the user has entered 1, "two" if the user has entered 2, and "three" if the user has entered 3. If a number other than 1, 2 or 3 is entered, the program should display an error message.

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

int main(){
        int userNum;

        cout << "Enter one of the numbers 1, 2, or 3: ";
        cin >> userNum;
        //
        // Write the switch statement here.
```

```
        //
        return 0;
}
```

g) [20 marks]

Convert the following `if/else if` statement into a `switch` statement:

```
if (choice == 1){
        cout << fixed << showpoint << setprecision(2);
}
else if (choice == 2 || choice == 3){
        cout << fixed << showpoint << setprecision(4);
}
else if (choice == 4){
        cout << fixed << showpoint << setprecision(6);
}
else {
        cout << fixed << showpoint << setprecision(8);
}
```

h) [40 marks]

Write a piece of C++ program that prompts the user for entering a number, say `x`, and prints:

```
1, 2, 3, …, x-1, x, x-1, …, 3, 2, 1
```

Assume that `x > 1`. For example, if the input is 5, the output should be 1 2 3 4 5 4 3 2 1, similarly if the input is 2, the output should be 1 2 1.

i) [20 marks]

Let `Nums` be an integer array with 20 elements. Write a `for` loop that prints each element of the array.

j) [10 marks]

Define a two-dimensional array of integers named `grades`. It should have 30 rows and 10 columns.

k) [20 marks]

An application uses a two-dimensional array defined as: `int days[29][5];`

Write code that sums each row in the array and displays the results.

l) [10 marks]

Look at the following array definition: `int set[10];`

Write a statement using `pointer` notation that stores the value `99` in `set[7].`

m) [20 marks]

Write code that dynamically allocates an array of `x` integers (where `x` is taken as input from the user), and then uses a loop to allow the user to enter values for each element of the array. Perform both the operation using a `pointer`.

n) [10 marks]

Assume that `tempNumbers` is a pointer that points to a dynamically allocated array. Write code that releases the memory used by the array.

o) [10 marks]

Look at the following function definition.

```
void getNumber(int &n){
        cout << "Enter a number: ";
        cin >> n;
}
```

In this function, the parameter n is a reference variable. Rewrite the function so that n is a pointer.

p) [20 marks]

Declare a structure named Circle, with the following members:

```
x: an integer
y: an integer
r: a float
```

Next, write statements that

A) define a Circle structure variable named circle

B) assign 12 to the x member of circle

C) assign 7 to the y member of circle

D) assign 3.5 to the r member of the circle

D) display the contents of the x, y and r members of circle

Q. No. 4 [100 marks]

Source Character	A	B	C	D	E	F	G	H	I	J	K	L	M
Encrypted Character	*L*	*M*	*D*	*P*	*A*	*R*	*T*	*F*	*V*	*G*	*Y*	*K*	*Z*

Source Character	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Encrypted Character	*O*	*N*	*C*	*E*	*Q*	*B*	*S*	*H*	*U*	*I*	*W*	*J*	*X*

(a) Write a function named `encrypt` that receives a string/c-string containing only the alphabets [A-Z] and its size as arguments and return the encrypted string. The encryption should be performed using the lookup (mapping) table given above. For e.g., `encrypt("PAKISTAN", 8)` should return `"CLYVBSLO"`.

(b) Write a function named `decrypt` that receives a string/c-string containing only the alphabets [A-Z] and its size as arguments and return the decrypted string. The decryption should be performed using the lookup (mapping) table given above. For e.g., `decrypt("CLYVBSLO", 8)` should return `"PAKISTAN"`.

Q. No. 5 [50 marks]

Write a complete C++ program that reads a 3x3 matrix as input, and computes and displays the trace and determinant of the matrix.

Recall that

(1) the trace of a square matrix is the sum of the diagonal values.

(2) the determinant of a 3x3 matrix is computed as shown below:

$$\begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = a(ei - fh) - b(di - fg) + c(dh - eg)$$

Q. No. 6 [50 marks]

Lets assume that we have two vectors:

$$\vec{V} = [v_1, v_2, v_3]$$
$$\vec{W} = [w_1, w_2, w_3]$$

The dot product between the two vectors is done as shown below: $\vec{V}.\vec{W} = v_1w_1 + v_2w_2 + v_3w_3$

The cross product between the two vectors is done as shown below: $\vec{V} \times \vec{W} = [v_2w_3 - v_3w_2, v_3w_1 - v_1w_3, v_1w_2 - v_2w_1]$

Write a complete C++ program that defines a structure to store a vector's information, then read two vectors as input and the display (1) the dot product, and (2) the cross product between the given vectors.

Note that a solution without using structures would not be graded.

Q. No. 7 [100 marks]

A minimax or saddle point in a two-dimensional array is an element that is the minimum of its row and the maximum of its column, or vice verse. For example, in the following array

11	22	33	33
99	55	66	77
77	44	99	22

the element 33 is a minimax because it is the maximum of row 0 and the minimum of column 2. The element 55 is another minimax because it is the minimum of row 1 and the maximum of column 1.

Write a program that reads a 4-by-3 matrix, and then prints the location and value of each minimax in the matrix. For example, it would print

a[0][2] = 33 is a minimax
a[1][1] = 55 is a minimax

for the matrix shown above.

Q. No. 8 [100 marks]

Write the following function:

```
void rotate(int a[], int n, int k)
```

The function "rotates" the first `n` elements of the array `a`, `k` positions to the right (or `-k` positions to the left if `k` is negative). The last `k` elements are "rotated" around to the beginning of the array. For example, if `a` is the array shown below:

Offset	0	1	2	3	4	5	6	7
Element	22	33	44	55	66	77	88	99

Then the call `rotate(a,8,3)` would transform `a` into

Offset	0	1	2	3	4	5	6	7
Element	77	88	99	22	33	44	55	66

Note that the call `rotate(a,8,-5)` would have the same effect.