Name: Model Answers Roll No:

0	1	2	3	4	5	6	7	Total

Indian Institute of Technology Mandi

IC150: Quiz 1

12th April 2013, 8:00-8:50 a.m.

Answer all questions. ***No calculators or cellphones***. Maximum marks: 25

0) When you are PM, what will be your highest priority (select one)? [½]
   a) Ban T20 matches on weekdays
   b) Double the number of T20 teams
   c) Double the number of IITs
   d) Double the salary of IIT faculty
   e) Other (specify):

1) Fill in the blanks: [7½]

   a) Expand the acronym FLOPS _Floating point operations per second_____

   b) `add ax, bx` is a statement in an _______assembly_____________________ language, whereas `a = b + c/d – sin(x)` is a statement in a __high level_ language.

   c) The difference engine was designed by _____Charles Babbage_____in 1882.

   d) The first electronic computer was named __ENIAC______________.

   e) Two programming paradigms are ___Imperative /OOPS/Object-oriented/ Logic/ functional/ procedural_ and ____.

   f) If a C function needs to return 2 or more results, it must use call-by-_Reference____

   g) The value of the C expression `7%-4` is _____3____________.

   h) The value of the C expression `5 && 6 * (3-13/4)` is _0/false________.

   i) The value of the C expression `(2)?3:4` is _3______.

   j) The range of values that can be stored in a 24-bit unsigned integer is _____0______ to ________2^{24}-1____(give your answer using 2^n notation).

   k) A Beowulf cluster consists of many _________Workstations or PCs_________as client nodes.

2) Consider the following C function that attempts to calculate the area of a rectangle given its length and height (both floating point). What changes will you make to ensure that the output value is correct and accurate? [2]

```
#include <stdio.h>
float RectArea(float len, float ht)
{
  float a;

  a = len * ht;
  return(a);
}
```

3) Answer briefly: [3]

(a) Give the value in base 2 of the C expression `0123 + 0xef`

```
   001010011
    11101111 
___101000010
```

(b) Assuming 32-bit integers and the declarations below, how many bytes of memory are occupied by the array A?

```
#define MAX_SIZE 20
#define NUM_YR 30
int A[NUM_YR*5][MAX_SIZE];
```

30 ×5×20×4 = 12,000 bytes

(c) What is a *sentinel*?

It is a special data value that indicates the end of a loop.

4) Hand-simulate the following code. In the table below, write the values of each variable after each iteration of the for loop, **before** the `printf()` is executed. [3]

```
int i, f, n = 10;
for(i = 0; i < 9; i++)
  {
    f = n%2;
    if (f == 0) n = n/2;
    else n = 3*n+1;
    printf("%d ", n);
  }
```

Iteration no.	**Value of i**	**Value of f**	**Value of n**
1	0	0	5
2	1	1	16
3	2	0	8
4	3	0	4
5	4	0	2
6	5	0	1
7	6	1	4
8	7	0	2
9	8	0	1

5) Do the following conversions (the subscript indicates the base; show your working). [3]
(a) $(111001)_2$ to decimal

1
8
16
32
57_{10}

(b) $(1011111100010101)_2$ to hexadecimal

By inspection of groups of 4 bits: bf15

(c) $(8027)_{10}$ to base 12 (duodecimal)

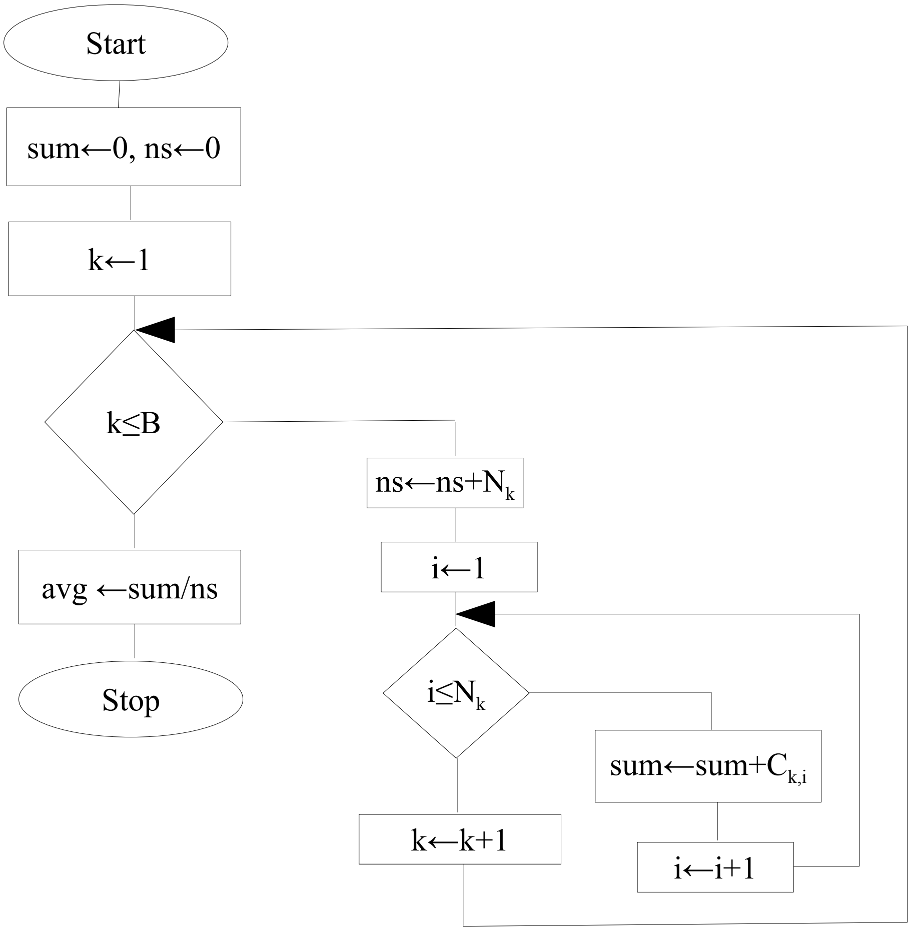
8027 = $478b_{12}$

6) What is the output of the program below? [2]

```
#include <stdio.h>
void DoOne(int a[], int n)
 {
   if (n)
    {
     printf("%d ", a[--n]);                54321
      DoOne(a, n);
    }
 }
int main()
 {
 int m[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
  DoOne(m, 5);m[0]                          [9]
 }
```

7) Design an algorithm to compute the average CGPA of all students in an Institute. The *B*-element vector **N** contains the number of students in each of *B* branches, where N_k is the number of students in branch *k*, $1 \le k \le B$. The CGPAs are stored in the 2-dimensional matrix **C**. The data for all students in branch *k* is stored in the row C_k, where $C_{k,i}$ contains the CGPA for the i^{th} student in branch *k*, $1 \le i \le N_k$.
Draw a neat flow-chart for your algorithm. Assume that the data is already available in **N** and **C**, and store the result in the variable *avg*. Do **NOT** write C code. [4]

```
1. sum ← 0, ns ← 0
2. For each branch k in 1 .. B do
2.1  ns ← ns + $N_k$
2.2  For each student i in 1 .. $N_k$ do
2.3          sum ← $C_{k,i}$ + sum
3  avg ← sum/ns
```


Start
sum←0, ns←0
k←1
k≤B
ns←ns+N_k
i←1
i≤N_k
sum←sum+$C_{k,i}$
i←i+1
k←k+1
avg ←sum/ns
Stop