ESC101A: Fundamentals of Computing
(Mid Semester Exam)

Department of Computer Science and Engineering, IIT Kanpur

September 22, 2017

Version B

Instructions

1. Write you name, section and roll number on all pages.
2. Please use pens (blue/black ink) and not pencils. Do not use red pens for answering.
3. Even if no answers are written, the answer book has to be returned back with name and roll number written.
4. Recall that cheating carries severe consequences.
5. Assume that necessary header files are included in all the program snippets.
6. All blanks in one question carry same weight.

Question	Points	Score
1	29	
2	20	
3	15	
4	25	
5	11	
Total:	100	

Name:	
Roll No:	
Section:	

Question 1. (a) (2 points) What is the output of the following program?

```
int main() {
  int c ;
  c = -15.3/4;
  printf("%d\n", c);
  return 0; }
```

Solution: -3

(b) (5 points) What is the output of the following program?

```
#define mul(x,y) x*y
#define mul2(x,y) (x)*(y)
int main() {
  int a =8;
  printf("%d ", 64/mul(a,2+6));
  printf("%d", 64/mul2(a,2+6));
  return 0; }
```

Solution: 22 64

(c) (6 points) **Select** True or False.

(a) (2 points) Float data type can store both positive and negative decimal numbers. (True / False)

(b) (2 points) Not passing arguments to a function defined as **int foo(int a)** in the caller program will give compilation error. (True / False)

(c) (2 points) In theory, function **sizeof()** may return both positive and negative values. (True / False)

Solution: True, True, False

(d) (4 points) What is the output of the following program?

```
int main() {
  int i=0, j=1, x;
  x = !i||j;
  while(j > 0) {
    j--;
    int x = !i&&j;
    printf("%d,", x);
  }
```

```
  printf("%d", x);
  return 0; }
```

Solution: 0,1

(e) (12 points) What is the output of the following program for the following input?
Input:
6
9 22 1 3 1 1

```
void bar(int a[], int n, int b) {
  int i;
  for (i = 0; i < n; i = i + 1) {
    a[i] = a[i] + b * (i + 1);
    b = b + 1;
  }
  for (i = 0; i < n; i = i + 1)
    printf("%d ", a[i]);
  printf("\n");
}

int foo(int a[], int n, int b) {
  int i, sum = 0;
  for (i = 0; i < n; i = i + 1) {
    sum = sum + a[i];
    a[i] = a[i] * b;
  }
  for (i = 0; i < n; i = i + 1)
    printf("%d ", a[i]);
  printf("\n");
  return sum; }

int main() {
  int i, n, a[10];
  scanf("%d", &n);
  for (i = 0; i < n; i = i + 1)
    scanf("%d", &a[i]);
  bar(a, n, foo(a, n, 6));
  return 0; }
```

Solution:
54 132 6 18 6 6
91 208 123 178 211 258

Name: Section: Rollno:

Question 2. (20 points) **Dr. Kaprekar's illegibility** (2*10 = 20 points)

Dr. Kaprekar defined Kaprekar number as a natural number whose square can be split into two parts that add up to the original number again.

For instance, 9 is a Kaprekar number, because $9^2 = 81$ and $8 + 1 = 9$.

The second part may start with the digit 0, but must be nonzero. For example, 999 is a Kaprekar number, because $999^2 = 998001$, which can be split into 998 and 001, and $998 + 001 = 999$.

But 100 is not; although $100^2 = 10000$ and $100 + 00 = 100$, the second part here is zero.

The mathematician had written down a code years ago which output **True** when a given number n was a Kaprekar number and **False** otherwise. But his pathetic handwriting prevents him from understanding a few words. Help the old man by **filling in the blanks** for him.

```
1  // returns 1 if m is a Kaprekar, 0 otherwise
2  int kap(int m) {
3      if (m == 1)
4         return 1;
5      int sq_m = m*m;
6      int digit = 0;
7      while ( sq_m >0 ) {
8          digit++;
9          sq_m = sq_m / 10;
10     }
11     sq_m = m*m;
12     for (int r_digit = 1 or 0 or digits/2; r_digit < or <=
          digit ; r_digit++){
13      /* pow(a,b) returns a^b */
14      int eq_part = pow( 10 ,r_digit);
15      if ( eq_part == m or sq_m%eq_part==0 )
16              continue;
17      int sum = (sq_m/eq_part) + (sq_m%eq_part);
18      if (sum ==m)
19        return 1;
20     }
21     return 0; }
22
23 int main() {
24    int m;
25    scanf("%d", &m);
26    kap(m)?printf("True"):printf("False");
27     return 0;
28 }
```

Question 3. (a) (4 points) What is the output of the following program?

```
int main() {
  int a=0,b=1,c,x=10,y=2,z=2;
  c = (b++)? b==1&&a:0;
  printf("%d\n",c);
  b=10;
  c=30;
  if(a--)
    b+=c++;
  else
    b+=--c;
  printf("%d,%d,%d",a,b,c);
  return 0; }
```

Solution:
0
-1,39,29

(b) (2 points) What is the output of the following program?

```
int main() {
  char a[] = {'E','s','c','\0','1','0','1'};
  printf("%s\n",a);
  return 0; }
```

Solution: Esc

(c) (4 points) What is the output of the following program?

```
int main() {
  int a=0,i,M=20,j;
  for(i=0; i<M+1; i++) {
    j = i;
    while(j<M){
      a++;
      j++;
    }
  }
  printf("%d\n",a);
  return 0; }
```

Solution: 210

(d) (5 points) What is the output of the following program for the following input?

Input:

10

0 10 10 25 11 13 42 20 20 9

```
int main() {
  int i, n, arr[100];
  scanf("%d", &n);
  for (i = 0; i < n; i = i + 1)
    scanf("%d", &arr[i]);
  i = 0;
  do {
    switch (arr[i]) {
      case 0:
        printf("Democracy is so overrated.\n");
        break;
      case 10:
      case 20:
      case 30:
        printf("Underwoood 2016!?\n");
        break;
      case 42:
        printf("The road to power is paved with hypocrisy
           , and casualties ;)\n");
      default:
        printf("Chaos is a ladder\t");
    }
  i = i + 2;
  } while (i < n);
  return 0; }
```

Solution:

Democracy is so overrated.
Underwoood 2016!?
Chaos is a ladder The road to power is paved with hypocrisy, and casualties ;)
Chaos is a ladder Underwoood 2016!?

Name: Section: Rollno:

Question 4. (a) (20 points) **Pascal's triangle** (2*10 =20 points)

Pascal's triangle is a triangular array of the binomial coefficients. Following code takes an integer value m as input and prints first m lines of the Pascals triangle. Number of entries in every line is equal to line number.

For example, m=5 should output:

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1

The first line prints $\binom{0}{0}$, the second line prints $\binom{1}{0}$ $\binom{1}{1}$ and so on.

Please **fill in the blanks** to complete the program.

```
1  int main() {
2    int n, line, i, j, coeff, k;
3    scanf("%d",&n);
4    for (line = 0; line <= n-1; line++) {
5      /* Every line has number of integers equal to line
          number*/
6      for (i = 0; i <= line; i++) {
7        coeff = 1;/*coeff stores the required coefficient*/
8        /*k stores minimum of line - i and i*/
9         if (i > line - i)
10          k = line - i;
11        else
12          k = i;
13        for (j = 0; j < k ; ++j) {
14           coeff *= line - j;
15           coeff /= (j + 1);
16        }
17        printf("%d ", coeff);
18     }
19     printf("\n");
20   }
21   return 0; }
```

(b) (5 points) **Legendre's Formula** (1*5 = 5 points)

Following code finds the **greatest** integer x such that given an integer n and a prime number p, p^x (p raised to power x) divides n! (factorial).For example, $n = 5$ and $p = 2$ should output 3. Please **fill in the blanks** to complete the program.

Hint: The required answer $= \lfloor \frac{n}{p} \rfloor + \lfloor \frac{n}{p^2} \rfloor + \lfloor \frac{n}{p^3} \rfloor + ...$

where $\lfloor x \rfloor$ is the Greatest Integer Function.

```
int main() {
  int n, p, x=0; /*Take input integer n and p*/
    scanf("%d,%d",&n,&p);
    while (n or n>0  or n>=p or n/p>=1) {
        n = n/p;
        x +=  n;
    }
    printf("The required value of integer x  is %d\n",  x
       );
    return 0; }
```

Question 5. (11 points) **Gangs of Sub-Arrays** (1*11=11 points)

It was a usual day in Wasseypur. Sardar Khan and Ramadhir Singh were having a heated argument. This time, they were unable to decide who amongst them is a better programmer. To test Ramadhir's abilities, Sardar gave him a tricky question: Given an array, find out the maximum possible sum any sub-array can have. A sub-array is any contiguous segment of the original array. E.g., for the array $\{-2, 1, -3, 4, -1, 2, 1, -5, 4\}$, $\{1, -3\}$ is a sub-array but $\{-3, 4, 2, 1\}$ is not a sub-array. The subarray with the largest sum is $\{4, -1, 2, 1\}$, with sum 6.

Sadly, Ramadhir was killed by Faizal Khan while writing the solution to the question. It was his dying wish that someone completes this solution after he is gone. Can you help in completing Ramadhir's dying wish?

```
int max(int a, int b) {
    return (a >= b) ? a : b; }

int maxSumSubarray(int a[], int n) {
   int answer = a[0];
   int curr_max = a[0];
   int i;
   for (i = 1; i < n; i = i + 1) {
        curr_max = max(a[i], curr_max+a[i]);
        answer = max(answer, curr_max);
   }
   return answer;
}

int main() {
   int a[] =  {-2, -3, 4, -1, -2, 1, 5, -3};
   int n = sizeof(a)/sizeof(a[0]);
```

```
18    printf("%d",  maxSumSubarray(a, n));
19    return 0; }
```