Department of Computer Science and Engineering
Indian Institute of Technology Patna

Course:CS101 Spring 2016 Mid Sem Exam
Time: 120 minutes Full Marks 100

Roll No: ---------- Name: ----------

Signature of Invigilator: ----------

Answer in the question paper itself. Rough work can be done in the supplementary sheets.

1. For each of the following program/function state the possible output. Assume that *stdio.h* is already included. $(10 \times 3) = 30$

(a)
```
int f(int x)
{
  return 3*x + 1;
}
int g(int x)
{
  return f(2*x) - f(x);
}
int main()
{
  printf("%d\n", f(g(2)));
  return 0;
}
```

Answer ----------

Justification:----------

(b)
```
int main()
{
  int i, a[10];
  a[0] = 1; a[1] = 1;
  for (i = 2; i < 10; i++)
  {
   a[i] = a[i-1] + a[i-2];
   printf("%d ", a[i]*a[i-2] - a[i-1]*a[i-1]);
  }
  printf("\n");
  return 0;
}
```

Answer: ----------

Justification:----------

(c)
```
int fn( int a, int b)
{
 if (b == 0)
  return b;
 else
  return (a * fn(a, b - 1));
}
int main()
{
 int a=30,b=50,x;
 x=fn(a,b);
 printf("%d",x);
 return 0;
}
```

Answer: ----------

Justification:----------

(d)
```
int main ( )
{
int a, b=0;
static int c [10]={10,11,20,25,70,16,41};
for(a=0; a<10;++ a)
if ((c[a]%3)== 0)
b+ = c [a];
printf("%d",b);
return 0;
}
```

Answer: ----------

Justification:----------

(e)
```
int main()
{
 int k=3,I=4,m;
 m=++k +I--;
 printf("Value of m %d\n",m);
 m=k++ + --I;
 printf("Value of m %d\n",m);
 return 0;
}
```

Answer: ----------

Justification:----------

(f)
```
#define N 10+10
int main()
{
 printf("%d",N*N);
 return 0;
}
```

Answer: ----------

Justification:----------

(g)
```
int main()
{
 int i;
 for (i=1; i<100; ++i)
   i *= i+1;
 printf("%d", i);
 return 0;
}
```
Answer: ----------

Justification:----------

(h)
```
int main()
{
 int a=1,b=1;
 if(--a && --b)
   printf("a:%d b:%d", a,b);
 else
   printf("a=%d b=%d", a,b);
 return 0;
}
```
Answer: ----------

Justification:----------

(i)
```
int eval ( int q )
{
 q *= 10;
 q -= 100;
 return q;
}
int main ()
{
 int p = 10, q = 100;
 q = eval(p);
 printf("%d %d ", p, q);
 q = eval(q);
 printf("%d %d\n", p, q);
 return 0;
}
```

Answer: ----------

Justification:----------

(j) What does fun(98789) will return?

```
unsigned int fun(unsigned int n)
{
  unsigned int m;
  m = 0;
  while (n > 0)
  {
    m = (m*10) + (n%10);
    n = n/10;
  }
  return m;
}
```

Answer: ----------

Justification:----------

---

2. For each of the following programs/program segments/functions, point out if there is any compilation or runtime error. In case there is no such error, mention the output. Assume that *stdio.h* is already included. $(10 \times 3) = 30$

(a)
```
int main()
{
char ch;
int i;
scanf("%c", &i);
scanf("%d", &ch);
printf("%c %d", ch, i);
return 0;
}
```

Answer: ----------

Justification:----------

(b)
```
int main()
{
int size, i;
scanf("%d", &size);
int arr[size];
for(i=1; i<=size; i++)
{
 scanf("%d", arr[i]);
 printf("%d", arr[i]);
}
return 0;
}
```

Answer: ----------

Justification:----------

(c)
```
int main()
{
 int res;
 res= 56>76 ? return 0:return 1;
 printf("%d",res);
}
```

Answer: ----------

Justification:----------

(d)
```
void message()
{
 printf("\nPraise worthy and
        C worthy are synonyms") ;
}

int main()
{
 message(message()) ;
 return 0;
}
```

Answer: _____

Justification:_____

_____

(e)
```
int main()
{
int temp ;
scanf ( "%d", &temp ) ;
switch ( temp )
{
case (temp <= 20):
printf ("\nOoooooohhhh! Damn cool!");
case (temp > 20 && temp <= 30):
printf("\nRain rain here again!");
case(temp > 30 && temp <= 40):
printf("\nWish I am on Everest");
default:
printf("\nGood old nagpur weather");
}
```

Answer: _____

Justification:_____

_____

(f)
```
int main()
{
  long int _=5l;
  printf("%ld",_);
  return 0;
}
```

Answer: _____

Justification:_____

_____

(g)
```
int main()
{
  int i=2;
  printf("%d", i=i++==3);
  return 0;
}
```

Answer: _____

Justification:_____

_____

(h)
```
int main ()
{
  int x = 0;
  if (x = 0) printf("Case (a): %d", x);
  else if (x -= 7) printf("Case (b): %d", x);
  else printf("Case (c): %d", x);
  return 0;
}
```

Answer: _____

Justification:_____

_____

(i)
```
int main()
{
   int k = 1;
   for (;k; k--)
     printf("Hello");
   return 0;
}
```

Answer: _____

Justification:_____

_____

(j)
```
int foo()
{
    return (1,2);
}
int main()
{
    int x = 0;
    x = foo();
    printf("%d", x);
    return 0;
}
```

Answer: _____

Justification:_____

_____

3. Complete the C program or function for each of the followings. Please fill up the parts indicated by dashed lines to make it work correctly. Each dashed line should preferably be filled with one C statement.

(a) This following C program gives the factorial of a value using recursion. (10)

```
int main( )
{

 int a, fact ;

 printf ("\nEnter any number " ) ;

 scanf ( "%d", &a ) ;


 fact = -------------------- ;
 /*function call*/

 printf ( "Factorial value = %d", fact ) ;

}

int rec(int x)
{

  int f;

  if (x == 1)


          ------------------------- ;
/*return value satisfying base condition*/


  else


 ---------------------------;
/*recursive function call*/


   ---------------------------------;
   /*return the final value to main function*/
}
```

(b) The following C function that takes a floating-point value x as its only argument and returns the rounded value of x. The rounded value of x is the integer nearest to x. When x is mid-way between two consecutive integers, we follow the convention "round half away from zero", that is, round(2.5) = 3 and round(-2.5) = -3.. (10)

```
int roundit ( double x )
{

  int r; /* The rounded integer to return */

  double fpart; /* Fractional part */


  ----------------------------------------;
  /*Store in r the truncated value of |x|*/


 ----------------------------------------;
  /*Store in fpart the fractional part of |x|*/


 if (fpart >= 0.5) ------------------------ ;
 /*Modify r based conditionally upon fpart*/


 return ---------------------------- ;
  /*Return r after sign adjustment*/
}
```

4. Write a C program or function as stated in the followings-

(a) Find the frequency of even and odd numbers in an $n \times m$ matrix. (10)

Answer:

(b) For a given array of integers, write a function in C that counts the number of subarrays (of size more than one) that are strictly increasing. The function should take two arguments- the array and the number of elements of the array and returns an integer that provides the number of subarrays that are strictly increasing. *For example, if Input: Array[] = {11, 28, 28, 32} then Output: 2 as there are 2 subarrays {11, 28} and {28, 32}* (10)

Answer: