

Course Title : Computer Programming (CS F111)

Component: Test I (Regular) (CB), Max. Marks: 45 marks (15%), Duration: 1 hour, Date : 22-02-2012 (10.30 to 11.30am)

ID No. :

Name:

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Total	Recheck

Instructions: 1. Overwritten answers and answers written in pencil will not be considered for recheck. 2. Calculator is not allowed. 3. In case there is an error in the program, write Error in the space provided. 4. Assume for Q1 and Q2, the program fragments are written inside a main function. For all the questions the relevant header files are included.

Q1. What is the output of the following statements?

- `printf("%d", 12^0);`
- `char a = 'z';
printf("%c", (a >= 'a' ? a-- : '!'));`
- `int a = b = c = 65;
printf("a = %c, b = %f, c = %d", a, b, c);`
- `char ch = 'a';
switch (ch){
 case 'a': printf("I am 'a'");
 break;
 case 'b': printf("I am 'b'");
 break;
 case 97: printf("I am 97");
 break;
}`
- `printf("j=%f", 25.0%5.0);`
- `int i = 10;
if(i = printf(""))
 printf("Does not matter");
else
 printf("It matters");`

a) 12	c) Error	e) Error
b) z	d) Error	f) It matters

(6x1=6marks)

Q2. What is the output of the following statements?

- `float Int = 3.0; printf("%d", (int)Int);`
- `int x=10; int y=1; int z=2;
x <<= y | z;
printf("%d", x);`
- `int v1 = 5, v2 = 6, v3 = 7, mix;
mix = v1 ? v3 ? v1 : v3 : (v2) ? v2 : v3;
printf("%d\n", mix);`

a) 3	b) 80	c) 5
------	-------	------

(3x2=6marks)

Q3. a) Convert 8 bit 2's complement binary number (11111110) to decimal number.

- Find the value of x. $(9)_{10} = (10)_x$

- Convert $(10110010010010)_2 = (?)_{16} = (?)_8$
- Following piece of code will give error when compiled. Add a SINGLE line that makes the code error free.

```
main(){
    NUM a = 15.0;
    printf("a=%f", a);
}
```
- There are 26 characters in the English alphabet. What is the least number of bits needed to give each character a unique bit pattern?
- In continuation to above how many more bits are needed to distinguish between lower and upper case versions of all 26 characters?

a)-2	c)(2C92) ₁₆ , (26222) ₈	e)5
b)9	d)typedef float NUM;	f)1

(6x1=6marks)

Q4. Find the output for the following programs:

- ```
main(){
 int x = 10, y = 100%90, i;
 for(i = 1; i < 10; i++)
 if(x != y)
 printf("%d,%d\n", x, y);
 printf("hello");
}
```
- ```
main(){
    int a = 2, b = 2, c = 3, d = 0;
    if(a < 2 || b == 2){
        printf("first\n");
        if(c > 2 && d++)
            printf("second");
        printf("third\n");
    }
    else
        printf("fourth");
}
```



```

main(){
int i=0;
for(;;i++){
    i++;
    printf("%d",i);
    if(i >= 10)
        break;
}
}

```

a)10,10 hello	b)first third
c)1357911	

(3x2=6marks)

Q5. Find the output for the following programs:

a) int num = 25;
main(){
{ int num = 20;
printf("%d ",num/5*5);
}
{ printf("%d ",num/5*5); }
}

b) main() {
int i;
for(i = 0; i < 3; i++)
switch(i) {
case 1: printf("%d ",i);
case 2 : printf("%d ",i);
default: printf("%d ",i);
}
}

c) main(){
int i = 6;
switch (i){
default : i+=2;
case 4: i = 4;
case 5: i++;
break;
} printf("%d",i);
}

a)20 25	b)0 1 1 1 2 2	c)5
---------	---------------	-----

(3x2=6marks)

Q6. a)The following program swaps the value of two variables without using a third variable. Fill the missing part with an expression.

```

main(){
int a = 10,b = 20;
a= (a+b)-(b=a);
printf("after swap\n a= %d \nb= %d\n",a,b);
}

```

b) How many times will **Hello** be printed in the following program?

```

main(){
int x;
for(x = -1; x <=10; x++){
    if(x < 5)
        continue;
    else
        break;
    printf("Hello\n");
}
}

```

Ans:0 times

(2x3=6marks)

Q7. Following partial program is used to generate a pattern which given below. Fill in the blanks to complete the program.

```

1
121
12321
1234321

```

```

main(){
int line = 4, space = line - 1, cnt = 2, i, j, k;
for(i = 1; i <= line ; i++){
    for(j = 1; j <= space; j++){
        printf(" ");
        space--;
    }
    for(k = 1; k <= 2 * i - 1; k++){
        if(k <= i)
            printf("%d", k);
        else{
            printf("%d", k-cnt);
            cnt+=2;
        }
    }
    printf("\n");
    cnt=2;
}
}

```

(5x1=5marks)

Q8. Which control structure (i. Sequential ii. Selection iii. Cascading Selection iv. Loop) would be most appropriate for the following problems:

- Checking for a correct input and continually rechecking if incorrect.
- Assigning grade to a student for the course CS F 111.
- Determining the volume of a sphere given a radius.
- Writing "Odd" or "Even" depending on an integer variable's value.

a)iv	b)iii	c)i	d)ii
------	-------	-----	------

(4x1=4marks)

Test II Answer Key

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE PILANI, K K BIRLA, GOA CAMPUS
Second Semester 2011-2012

Course Title : Computer Programming (CS F111)

Component: Test II (Regular) (CB) , Max. Marks: 45 marks (15%), Duration: 1 hour. Date : 30-03-2012 (10.30 to 11.30am)

ID No. : _____ **Name:** _____

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Total	Recheck

Instructions: 1. Overwritten answers and answers written in pencil will not be considered for recheck. 2. In case there is an error in the program, write **Error** in the space provided 3. Assume for Q1(a-d), the program fragments are written inside a main function. For all the questions the relevant header files are included. 5. Assume size of char is 1 byte, int is 2 bytes and address is 2 bytes

Q1 What is the output for the following program fragments? (1.5x6 = 9 Marks)

- a) `char dummy[80];
scanf("%c", dummy);
printf("%s", dummy);`
 What is printed if the user input for the above program fragment is: **Pilani campus**
- b) `int arr[10]={10};
printf("%d\n", 2[arr]);`
- c) `char s[] = "BITPILANIGOACAMPUS";
printf("%d", *(s+strlen(s)));`
- d) `char str[] = "Cat";
str[0] = 'B';
printf("%s", str);
str = "Cat";
printf("%s", str+1);`
- e) `void disp (int a, int b){
return(a > b ? a+b: a*b);
}
main(){
printf("%d", disp (5, 6));
}`
- f) `char word[]="fire";
int main(){
word[0]='h';
word[2]='\n';
printf("%s", word);
}`

a) Pil	b) 0	c) 0
d) Error	e) Error	f) hi e

Q2 What is the output for the following program fragments? (2x4 = 8 Marks)

- a) `int func1(int);`

```
main(){
    int k = 35;
    k = func1(k = func1(k = func1(k)));
    printf("k=%d ", k);
}
```

```
int func1(int k){
    return (k++);
}
```

b) `static int i = 4;
main{
char str[] = "INDIAN";
static int i;
while (str[i] != '\0'){
if (i%2 == 0)
printf("%c", str[i]);
i++;
}`

c) `main(){
int print();
int c = 0;
c = print();
printf("%d", c);
}
int print(){
return printf("Print");
}`

d) `void increment(int i){
i++;
}
main(){ int i;
for(i = 0; i < 2; increment(i))
printf("i=%d\n", i); }`

a) 35	b) IDA
c) Print5	d) Infinite Loop

**Q3 Fill in the blanks so that the output is
1 3 7 13 21... for array contents 1,4,9,16,25...
(6 Marks)**

```
void sequence (int *, int);

main(){
    int a[] ; /* Assume that the array is initialized with
               a[] = 1,4,9,16,25.....*/

    int *k, i, len;
    /* len holds the number of elements in a[]*/
    len = sizeof (a)/sizeof(int) ;
    sequence ( a, len );
    k = a ;
    for(i = 0; i < len; ){
        printf("%d ", *(k+i));
        i++;
    }
}

void sequence (int *ptr, int d){
    int x;
    for(x = 0; x < d; x++){
        *ptr++ = *ptr - x;
    }
}
```

**Q4 What is the output for the following program fragments?
(3x2 = 6 Mark)**

```
a) main(){
    char a[]="Visual c++";
    char *b="Visual c++";
    printf("%d,%d,", sizeof( a ), sizeof( b ) );
    printf(" %d ", sizeof( *b ) );
}

b) main(){
    int myarr[2][4] = {0, 1, 2, 3, 4, 5, 6, 7};
    printf("%d ",*(myarr[1] ));
    printf("%d ",*(1[myarr]+1) );
    printf("%d ",*(*(myarr+1)+2) );
}
```

a)	b)
11, 2, 1	4 5 6

**Q5 What is the output for the following program fragments?
(4+4 = 8 Marks)**

```
a) main(){
    int x[] = {1,6,9,7,5}, *p = x;
    printf("%d ", *++p+1);
    printf("%d ", ++*(p+1));
}

b)
#define var 3
main(){
    char *name[var]={"king","queen","prince"};
    char *ptr=name[1];
    puts(ptr++);
    puts(++ptr);
}
```

a) 7, 10	b) queen een
----------	-----------------

**Q6: Following program is used to find "key" in a sorted array using binary search key technique. Fill in the blanks using appropriate statements.
(8 Marks)**

```
main()
{
    int *begin, *last, *mid, key;
    int a[8]={11,22,33,44,55, 66,77,88};
    printf("Enter Key element: ");
    scanf("%d",&key);

    begin = &a[0]; /* a */
    last = &a[8-1]; /* a+7 */

    while( begin <= last ){
        mid = begin + (last-begin)/2 ;
        if ( key == *mid ){
            printf("Key found");
            break;
        }
        else if ( key > *mid )
            begin = mid + 1 ;
        else
            last = mid - 1 ;
    }
    if (begin > last )
        printf("Not found");
}
```


Course Title : Computer Programming (CS F111)

Component: Comprehensive (Regular) (CB)

Max. Marks: 90 marks (30%)

Duration: 3 hours

Date : 08-05-2012 (2.00 to 5.00p.m)

ID No. :

Name:

Lab sec no:

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Total	

Instructions: 1. Overwritten answers and answers written in pencil will not be considered for recheck. 2. In case there is an error in the program, write **Error** in the space provided. 3. Assume for Q2a,b,d,e. the program fragments are written inside a main function. 4. For all the questions the relevant header files are included. 5. Assume all files will be opened successfully. 6. Assume size of char is 1 byte, short int is 2 bytes, int is 4 bytes and address is 4 bytes. 7. **Calculator is not allowed.** 8. ASCII code of 'A' is 65 and 'a' is 97.

Q1. [2+2+2+3+3+3=15M]

a) How many number of ones are there in unsigned binary representation of the number

$$16^3 * 9 + 16^2 * 7 + 16 * 5 + 3?$$

9

b) If x is number in Q1a. and y is $16^3 * 2 + 16^2 * 4 + 16 * 6 + 8$, then

$$x | y = (\quad 133573 \quad)_8$$

c) The 2's complement representation of the

decimal value -16 is

10000

Use minimum number of bits for representation.

d) Fill in the blank such that output of the code is 3,2. Assume that the architecture follows Little Endian byte ordering and each address stores one byte.

```
main(){
    union xyz{
        char ch[2];
        short int n;
    }a;
    a.n = 515 ;
    printf("%d,%d", a.ch[0], a.ch[1]);
}
```

e) Represent 29.25 in IEEE 754, single precision format. Express the answer in hexadecimal.

41EA0000

f) Find the decimal equivalent of single precision representation of IEEE floating point number 0xC1468000.

-12.40625

Q2. Find the output. [2+2+2+4+5=15M]

a) enum color{red, blue=2, pink} var = red; printf("%d %d", var, sizeof(var));

b) char str1[20] = "he", str2[20] = "she"; strcpy(str2, strcat(str2, str1)); printf("%s%s", str1, str2);

c) For the following sample input:
Jack
Jill

what is the output?

```
char * getname() {
    static char name[30];
    scanf("%s", name);
    return name;
}
main(){
    char *name1, *name2;
    name1 = getname();
    name2 = getname();
    printf("%s %s", name1, name2);
}
```

d) int a = 3, b = 9; double x = 12.335; printf("%d\t", a += (a-b) ? (++a) : (--b)); printf("%6.*lf\t %6.2f", 3, x, x);

2.a 0 4

2.b heshehe

2.c Jill Jill

2.d 8 12.335 12.34 or 12.33

c)

```

char a[] = "Goa campus";
char *b = (char *) malloc (strlen (a) + 1);
char *p, *r;
strcpy (b, "Goa campus");
p = b + 1;
r = b + 4 - (b - p);
printf("%s ", r);
printf("%d %d ", sizeof(a), sizeof(b));
printf("%d %d", sizeof(*a), sizeof(*b));

```

2c. campus 11 4 1 1

Q3a. What is the output of following code? [3M]

```

main(){
    int i, j; char y = '_', z = '*';
    for (i = 4; i >= 1; printf("\n"), i--) {
        for (j = 1; j <= i; j++) {
            if (i == 4) printf("%c ", z);
            else if (j == i) /* line no. 6 */
                printf("%c", z);
            else printf("%c", y);
        }
    }
}

```

Q3b. else if (j == i) condition in sixth line (refer Q3a) is changed to **else if (j = i)**, what will be the output? [2M]

3a	3b
* * * *	* * * *
- - *	*
- *	*
*	*

Q3 c. Fill in the blanks:[5M]

- All the static variables declared within main() are allocated space on heap part of memory.
- Dynamically allocated memory can be referred using pointer
- The symbolic constant EOF has ASCII value 26
- free() function returns dynamically allocated memory to the heap.
- The dynamic memory allocation functions are defined in stdlib.h header file.

Q4 a. Fill in each box with **only one statement** to complete the compress() function which

removes blank spaces in the given source string. Pointer s points to the starting address of the source string.[2M+2M]

```

char * compress(char *s){
    char *p, *ps;
    for (ps = p = s; *s != '\0'; s++){
        if ((*s) != ' ')
            *p++ = *s;
    }
    *p = '\0';
    return(ps);
}

```

4b. What is the output? [3M + 2M]

```

int fun(int y){
    static int i;
    if(y==1 || y==3)
        return y;
    else
        for(i = 0; i < 3; i++, y *= 8)
            return y;
}

void main(){
    unsigned char a = 'A'; signed char b = 'B';
    int x[sizeof(a+b)];
    int i, val, y[] = {3, 5, 8, 3};
    int len = sizeof(y);
    int size = sizeof(x);
    for (i = 0; i < len; i++){
        val = fun(fun(y[len-16+i])+fun(y[i+3]));
        printf("%d %d", val, size);
    }
}

```

4b. 6 16

5a. What is printed on the screen? [2M]

```

main(){
    FILE *fp1;
    unsigned char ch;
    fp1 = fopen("file1.txt", "r");
    printf("output:");
    while(!feof(fp1)){
        fseek (fp1, 0L, 0);
        printf("%c", ch = getc(fp1));
    }
    fclose(fp1);
}

```


Compre Answer key

Assume the contents of file file1.txt is BITS.

Q5a. output:

Q5b. Assume that the contents of file1.txt is **pqr** and contents of file2.txt is **xyz**. Answer the following questions with respect to the code given below:

```
main(){
    FILE *fp1, *fp2; char ch;
    fp1 = fopen("file1.txt", "r");
    fp2 = fopen("file2.txt", "a+");
    while (!feof(fp1)){
        ch = getc(fp1) - 32;
        fprintf(fp2, "%c", ch);
    }
    fseek(fp1, -1L, 1);
    printf("%c", getc(fp1));
    fseek(fp1, -2L, 2);
    printf("%c", getc(fp1));
    fclose(fp1);
    fclose(fp2);
}
```

i. Show the **contents** of file1.txt and file2.txt after executing the program. Also show the **positions** of file pointers fp1 and fp2 before closing the files (indicate using arrows) [2M+2M]

ii. What is displayed on the screen? [2M]

Q5b) i		Q5b) ii
file1.txt pqr ↑ fp1	file2.txt xyzPQR ↑ fp2	rq

Q6. Consider the code below and answer the following questions.

```
struct test{
    char arr[5];
    int i;
};
void fun1(struct test *s, char c){
    if(s->i == 4)
        printf("Array full");
    else{
        s->i++;
        s->arr[s->i] = c;
    }
}
```

```
void fun2(struct test *s){
    char c;
    if(s->i == -1)
        printf("Array empty");
    else
        c = s->arr[s->i--];
    printf("%c", c);
}
main(){
```

```
    struct test T;
    T.arr[0] = 'A';
    T.arr[1] = 'B';
    T.i = 1;
    fun1(&T, 'C');
    fun2(&T);
    fun2(&T);
    fun2(&T);
}
```

a. What is the output? [4M]

b. If main function in the above program is changed to

```
main(){
    struct test T;
    T.arr[0] = 'b';
    T.arr[1] = 'c';
    T.i = 1;
```

Function calls

write minimum function calls in proper sequence to get the output as **bcba** [6M]

6 a. CBA

6 b.

```
fun1(&T, 'b');
fun2(&T);
fun2(&T);
fun2(&T);
fun1(&T, 'a');
fun2(&T);
```

Q7a. Find the output. [3M]

```
int d[4] = { 3, 89, 47, 50 };
void f(int d[], int n){
```

```

if (n != 0){
    printf("%d ", d[--n]);
    f(d + 1, n);
}
}
main(){
    f(d, 4);
}

```

7a: 50 50 50 50

Q7b. Following function bsearch() finds key element in an array (array elements are stored in descending order) using recursive binary search method and returns the position of the key element if found else returns -1. Assume that initially low and high points to first and last elements respectively.

i) Fill in the missing statements [3M]

```

bsearch ( int a[], int key, int low, int high ){
    int mid;

```

```

    if (  )

```

```

        return (-1);

```

```

    mid = ( low + high ) / 2;

```

```

    if ( key == a[mid] )

```

```

        return ( mid );

```

```

    if ( key < a[mid] )

```

```

        return bsearch ( a, key, mid + 1, high);

```

```

    else

```

```

        return bsearch ( a, key, low, mid -1);

```

```

}

```

ii) For the above function, following array contents {12,11,9,5,3,2,-1,-5} are passed. How many times the function bsearch() will be called including the call from main to search an element -7. [3M]

7b) ii. 5 times

Q8. Answer the following questions. (Note: L1 to L4 labels are used just to point certain statements in the code.)

```

#define SIZE 3
struct student{
    int id;
    char *name;
};
typedef struct student s;
main(){
    int i, num;
    char fname[20];
    int len;
    s *p[SIZE];
    printf("Enter id and name: \n");
    for(i = 0; i < SIZE; i++) {

```

L1: p[i] = (s *)malloc (sizeof(s));

```

scanf("%d %s",&num, fname);

```

```

p[i]→id=num;

```

L2: len = strlen (fname);

```

p[i]→name = (char *) malloc (len +1);

```

```

strcpy(p[i]→name,fname);

```

```

}

```

L3: printf("size of %d,%d\n",sizeof(p), sizeof(p[0]→name));

```

/* L4: code to deallocate the memory */

```

```

for (i = 0; i <SIZE; i++){

```

```

    free (p[i]→name);

```

```

    free (p[i]);

```

```

}

```

```

}

```

- Write C statement at L1 to dynamically allocate memory to each array element of p. [3M]
- Write minimum number of statements at L2 to allocate memory dynamically for structure variable name. [4M]
- What is the output after executing L3. [2+2=4M]

12 , 4

- Write code in box(L4) to deallocate all the dynamically allocated memory. [3M]