

Solution/Model Answer of IAT-1 (March 2017)
Programming in C & data Structure – 15PCD23
II sem – All
Dr. P. N. Singh, Professor(CSE)

1. Explain Structure of a C Program. Write a program to calculate simple interest

10

Ans:

Structure of C Program

```

/* Separate line comments –
   ----- */
pre-processor directives /* header section */
global declarations
int main( )          /* main ( ) function section */
{
    local variables to function main ;
    statements associated with function main ;
    .....
    return (0);
}
udf1( )             /* user defined functions */
{
    local variables to user defined function udf1 ;
    statements associated with udf1 ;
}
.
.

/* program to calculate simple interest */
int main( )
{
    float p,r,t,si;
    printf("Enter principle, rate and time : ");
    scanf("%f%f%f", &p, &r, &t);
    si=p*r*t/100.00;
    printf("Simple interest = %f\n", si);
    return (0);
}

```

2.

a. Define an operator? Explain with Example

6

C includes a large number of operators. Individual constants, variables, array elements and function references can be joined by various operation to form expressions. The data items that operators act upon are called operands.

Examples of some operators:

Arithmetic Operators + (addition) - (subtraction) * (multiplication)
 / (division) % (modulus)

unary operator - (minus) acts upon a single operand
 (oh! C does not have an operator for exponentiation)

Relational Operators < (less than) <= (less than or equal to) > (greater than)
 >= (greater than or equal to)
 equality operators == (equal to) != (not equal to)

Logical Operators

&& (and) || (or)

unary operator ! negates the value of a logical operator

!(n<=10) - n is not less than or equal to 10

b. Describe data types in C? Explain the syntax for variable declaration and initialization. 4

<u>Data type</u>	<u>Description</u>
int	integer quantity
char	single character
float (real)	floating point number(i.e., a number containing a decimal point and/or an exponent
double	double-precision floating-point number (i.e., more significant figures, and an exponent which may be larger in magnitude.)

Syntax for variable declaration & initialization:

datatype variable;

int num;

float radius;

datatype variable = value;

int num=20;

float radius = 5.067;

3.

a. Define an Identifier? List the rules to construct an identifier? Give examples of valid and invalid identifier. 4

Ans:

Identifiers - Names of variables, function, or arrays are identifiers.

Rules:

→ They may contain A-Z, a-z, and _ (underscore) (no blanks)

→ An identifier name but must be started with an alphabet or _ (specific purposes)

→ Maximum length of name of identifier should be 31 characters (ANSI C standards) however in newer versions length of identifier name may be more.

→ Keywords & compiler constants should not be identifier name

Examples of valid identifiers: amount, order_no, marks1, stud[10], sort()

Examples of invalid identifiers: 1stpaper, **for**, **if**, name of student

b. Explain different conditional decision making statements with syntax. 6

Ans:

if statement:

if (expression/condition)

{

 statement1;

 statement2;

 ...

}

if else statement:

if (expression/condition)

{

 statement1;

 statement2;

}

else

{

 statement1;

```
    statement2;
    .....
}
```

Ladder of if ...else statement:

```
if (expression/condition)
{statement1; statement2; ..... }
else if (expression/condition)
    {statement1; statement2; ..... }
    else if (expression/condition)
        {statement1; statement2; ..... }
    .....
    else
        {statement1; statement2; ..... }
```

Nested if else statement:

```
if (expression/condition)
{ if (expression/condition)
    {statement1; statement2; ..... }
  else
    {statement1; statement2; ..... }
}
else { if (expression/condition)
    {statement1; statement2; ..... }
      else
        {statement1; statement2; ..... }
    }
```

switch statement:

```
switch (expression)
{
    case condition1
        statement1;
        statement2;
        .....
        break;
    case condition2
        statement1;
        statement2;
        .....
        break;
    .....
    default:
        statement1;
        statement2;
        .....
}
```

4.

a. Write a program to find greatest of the 3 numbers using nested if-else.

5

Ans:

```
/* program to find greatest in the 3 numbers using nested if-else */
#include <stdio.h>
int main( )
{
    int n1, n2, n3,max;
    printf("Enter 3 numbers : ");
    scanf("%d%d%d", &n1, &n2, &n3);
    if(n1 > n2)
        if(n1 > n3)
            max=n1;
        else
            max=n3;
    else
        if (n2 > n3)
            max=n2;
        else
            max=n3;
    printf("Greatest = %d\n", max);
    return (0);
}
```

b. Write a program to find whether the given year is leap or not. (consider century year)

5

Ans:

```
/* to find whether the given year is leap year or not */
#include <stdio.h>
int main( )
{
    int year;
    printf("Enter year : ");
    scanf("%d", &year);
    if((year%4==0 && year%100 != 0)|| year%400==0)
        printf("Yes, %d is leap year.\n", year);
    else
        printf("No, %d is not leap year.\n", year);

    return (0);
}
```

5.

a. Differentiate between while and do-while loop.

6

Ans:

The *while* statement:

```
while (condition)
{
    statement1;
    statement2;
    .....
}
```

The *do while* statement:

```
do
{
    statement1;
    statement2;
    .....
} while (condition);
```

- ➔ *While* is a pre-tested loop/entry controlled loop and *do... while* is post-tested loop
- ➔ Post-tested loop must execute at least once because condition is checked at the end of loop

- b. Write a program to check whether the given number is even or odd. (Use conditional Operator) 4

Ans:

```
/* to check whether a given number is even or odd - using
conditional operator*/
#include <stdio.h>
int main()
{
    int n;
    printf("Enter number : ")
    scanf("%d", &n);
    printf("Number is %s\n", n%2 == 0 ? " Even" : "Odd");
    return (0);
}
```

6.

- a. Evaluate the following expression. 4

```
a=5, b=6, c=7
d= ++a * b -c;
e = a & b
f=100%20<=20-5+100%10-20= =5>=1!=20;
g=c-- *b+d;
```

Ans:

d = 6 * 6 - 7	= 29
e = 6 & 6	= 6 (by bit wise operator 110 & 110)
f	= 1 (true, finally not equal to 20)
g = 7 * 6 + 29	= 71 (post decrement of c after statement)

- b. What will be value of x in following segments? 4

```
int a,b; float x; a=4; b=5; x=b/a;
int a,b; float x; a=4; b=5; x= (float) b/a;
```

Ans:

```
int a,b; float x; a=4; b=5; x=b/a;
x = 1.0 (integer values b/a is not casted)
int a,b; float x; a=4; b=5; x= (float) b/a;
x = 1.25 (b/a is casted)
```

- c. What is difference between 7, 7.0, '7', "7" ? 2

Ans:

7	Integer constant
7.0	floating point constant
'7'	character constant
"7"	string constant

7. Explain the syntax of switch case. Write a program to perform basic arithmetic operations like addition, subtraction, multiplication & division, using switch statement. (Error message should be displayed if number is divided by 0) 10

Ans:

```

/* Arithmetic operation by switch */
int main( )
{

    float n1,n2,result;
    int choice;
    printf("Enter 2 number : ");
    scanf("%f%f", &n1, &n2);
    printf("1. Addition 2. Subtraction 3. Multiplication 4. Division:\n");
    printf("Enter your choice number : ");
    scanf("%d", &choice);
    switch(choice)
    {
        case 1 : result = n1+n2;
                break;
        case 2 : result = n1-n2;
                break;
        case 3 : result = n1*n2;
                break;
        case 4 : if(n2 == 0)
                    { printf("divide error!!!\n"); return (99); }
                else
                    result = n1/n2;
                break;
        default: printf("Wrong choice!\n");
    }
    printf("Result = %f\n", result);
    return (0);
}

```

8. a. Write a program and pseudo-code to check whether the given number is palindrome or not. 5

Ans:

Pseudo-code to check given number is palindrome or not

```

START
INPUT NUM
TEMP=NUM
REV=0
WHILE TEMP <> 0
    REM=TEMP MOD 10
    REV=REV*10+REM
    TEMP=(INTEGER)TEMP/10

```

```

ENDWHILE
IF REV = NUM THEN
    PRINT "NUMBER IS PALINDROME"
ELSE
    PRINT "NUMBER IS NOT PALINDROME"
ENDIF
END.

```

```

/*Program to check given number is palindrome */
#include <stdio.h>
int main()
{
    long int num,temp,rev,rem;
    printf("Enter Number : ");
    scanf("%ld", &num);
    temp=num;
    rev=0;
    while (temp !=0)
    {
        rem=temp%10;
        rev=rev*10+rem;
        temp/=10;
    }
    if(rev==num)
        printf("Yes, %ld is palindrome\n", num);
    else
        printf("No, %ld is not palindrome\n", num);
    return (0);
}

```

b. Write a program to print Fibonacci series up to n terms.

5

Ans:

```

/* to print Fibonacci series up to n terms */
#include <stdio.h>
int main( )
{
    long int a=0,b=1,c;
    int n,x;
    printf("How many numbers of Fibonacci series ( > 2 ) ? ");
    scanf("%d", &n);
    printf("%7ld%7ld", a,b);
    for(x=3;x<=n;x++)
    {
        c=a+b;
        printf("%7ld", c);
        a=b;
        b=c;
    }
    return (0);
}

```