

Solution/Model Answer to IAT-1
 C programming for problem Solving – 18CPS23
 Dr. P. N. Singh, Professor(CSE)

1 a. Write 4 generations of computer with device, speed, size and example.

5 Marks

Ans:

Computer Generations

Generation	Duration	Devices	Purpose	Size & Speed	Examples
First	1940-50	Vacuum Tubes & Plug boards	General Purpose Electromechanical Systems	Huge & Very Slow	ENIAC, UNIVAC-I
Second	1950-60	Transistors/Semi conductors	Batch processing, Punched cards, Magnetic Tape	Little Big & slow	IBM 1401, MARK-III, UNIVAC 1107
Third	1960-70	IC(Integrated Circuits)	Parallel Processing, OS to manage Hardware, software & resources	Smaller & faster, Storage devices	IBM 360
Fourth	1970 onwards	LSI (Large Scale Integration)	GUI, Microprocessor, Networks & Internet	Small & very fast, Large storage capacity in MB	Intel C4004 8085, 8086, 80386 & 80486 Desktop PCs, Main Frame & Super computers
	Going on	VLSI	GUI & Network OS Ubuntu, , Android	Very small but Very-very Faster, Storage in Terabytes	Pentium I, II, III, IV, Dual core Laptops, Tablets, Smart Phones

Q 1b. Explain types of computer according to purpose and size

5 Marks

Ans:

According to purpose:

- a. **Analog computers:** Analog computer is a form of computer that is used to read & measure the physical phenomena such as electrical, mechanical, or hydraulic quantities (**Not discrete**) . Eg. Thermometer, Speedometer, Morse code Telex Machine, amplifiers etc.
- b. **Digital Computer:** A computer that performs calculations and logical operations with quantities represented as digits, usually in the binary number system. Example: Calculator, laptops, PCS
- c. **Hybrid Computer:** A combination of analog and digital computers those are capable of inputting and outputting in both digital and analog signals. A hybrid computer system setup offers a cost effective method of performing complex simulations. **Here calculations are done digitally and**

actions are taken mechanically or vice-versa. Super computers and robots are hybrid computers.

According to size:

- a. Super Computer: They are very large and very fast & task specific computers used by large organizations. These computers are used for **weather forecasting, meteorology, earth-quack studies, research and exploration** purposes. Super computers may be installed in satellites. NASA uses supercomputers for launching space shuttles, controlling them and for space exploration purpose. Example: **Titan, Cray(of India)**
- b. Mainframe Computer: These are smaller and less faster than super computers. Large firms & government organizations like banks, railways use Mainframes to run their business operations in their head/central office as a central role to update day to day transactions. Using virtual machines, mainframes run the various operating systems as if they were running on different computers. Example z9, z10
- c. Mini Computers: Minicomputers are used by small businesses & firms. Minicomputers are also called as "Midrange Computers". Individual departments of a large company or organizations use Mini-computers for specific purposes. Example K202, IBM Midrange
- d. Micro Computers: Desktop computers, **laptops, PCs, personal digital assistant (PDA), tablets & Smartphones, Scientific Clculators, Notebook, Gaming console, Sound and navigation System** are all types of microcomputers. The micro-computers are widely used & the fastest growing computers. Well known manufacturers are Dell, Apple, Samsung, Sony etc.

Q 2. Define an algorithm. Write an algorithm & program to find area of triangle by given 3 sides with proper alert message. 10 marks

Ans:

Algorithm: An algorithm is step-by-step (in the sequence) procedure/ formula to get the solution of a task. Number of steps must be finite. Criteria of an algorithm to reduce the complexity of the solution.

Algorithm to find area of a triangle by given 3 sides

```
Step 1.    Start
Step 2.           Input sides a,b & c
Step 3.    s=(a+b+c)/2
Step 4.    If s <= a OR s <= b OR s <= c then
    4.1           Print 'Wrong measurement of triangle'
Step 5.           Else
    5.1           area=sqrt(s*(s-a)*(s-b)*(s-c))
    5.2           Print area
Step 6.           Endif
Step 7.           End.
```

```
/* Program - Area of a triangle by given 3 sides*/
#include <stdio.h>
#include <math.h>
int main()
{
    float a,b,c,s,area;
```

```

printf("Enter 3 sides of a triangle : ");
scanf("%f%f%f", &a, &b, &c);
s=(a+b+c)/2.0;
if(s<=a||s<=b||s<=c)
printf("Invalid measurement: sum of 2 sides must be greater than 3rd\n");
else
{
    area=sqrt(s*(s-a)*(s-b)*(s-c));
    printf("Area = %.3f square unit\n",area);
}
return (0);
}

```

Expected output:

Enter 3 sides of a triangle : 4 5 12

Invalid measurement: sum of 2 sides must be greater than 3rd

Enter 3 sides of a triangle : 4 3 2

Area = 2.905 square unit

Q.3 a. Explain syntax with examples of printf() and scanf() statements.

4 marks

Ans:

printf() – Library function for formatted output:

Output data can be written on to a standard output device using the library function printf(). The printf statement provides certain features that can be effectively exploited to control the alignment and spacing of printouts on the terminals. The general form of printf statement is:

printf ("control string", arg1,arg2,, argn);

Control string of printf function consists of three types of item :

- Character that will be printed on the screen as they appear.
- Format specification that define the output format for display of each item.
- Escape sequence characters such as \n,\t, and \b.

scanf() – Library function for formatted input

Formatted input refers to an input data that has been arranged in a particular format. Input data can be entered into the computer from a standard input device by means of the C library function scanf. In general terms, scanf function is written as

scanf ("control string", arg1, arg2,, argn);

The control string specifies the field format in which the data is to be entered and the arguments arg1,arg2.....,argn specify the address of locations where the data is stored. Actually arguments are pointers which indicate where the data items are stored in computer's memory.

Control string contains field specification which direct the interpretation of input data. It may include:

- Field (or format) specification, consisting of the conversion character %, a data type character (or type specifier, and an optional number, specifying the field width.
- Blanks, tabs, or new lines.

- The arguments are written as variables or arrays, whose type match the corresponding character groups in the control string. Each variable name must be preceded by an ampersand (&). Array names should not begin with an ampersand.

Q 3 b. Describe Bit-wise operators & ternary (conditional if) operators with example.

6 marks

Ans:

& - Bit-wise AND : 1 only when both inputs are 1
 printf("%d", 20 & 25); /*10100 & 11001 = 10000 will give output 16*/
 | - Bit-wise or : 1 when any of the input is 1. It gives 0 when both inputs are 0
 printf("%d", 20 | 25); /*10100 | 11001 = 11101 will give output 29*/
 ^ - Bit wise XOR (Exclusive or): 1 when only one input is 1. If both are 1 or 0, it gives 0
 printf("%d", 20 ^ 25); /*10100 ^ 11001 = 01101 will give output 13*/
 ~ - Bit-wise compliment : Difference in highest & given. In Binary 1 will be 0 and 0 will be 1. If unsigned int has 2 bytes (16 bits) size with highest value 65535 then
 printf("%u", ~ 5); /*~ 0000000000000101 = 1111111111111010 will give output 65530*/
 >> - Bit wise right shift : Halves the value in integer
 printf("%u", 10>>2); /*1010 will 101 in first right shift & again 10 in 2nd right shift so 2 */
 << - Bit wise left shift : doubles the value(within range)
 printf("%u", 11<<2); /*1011 will 10110 in first left shift & again 101100 in 2nd left shift so 44 */

Ternary Operator(Conditional if): ? :

Syntax:

<condition> ? <statement for true> : < statement for false>

int salary = 50000;

Example1:

bonus = salary > 40000 ? 10000 : 20000;

Here salary is greater than 40000 so, bonus will be 20000

Q 4. Explain switch case statement with syntax and example of a calculator program to do simple operations like addition, subtraction, multiplication and division.

10 Marks

Ans:

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

```

                statement1;
                statement2;
                .....
            }

/* Arithmetic operation by switch */
int main( )
{
    float n1,n2,result;
    int choice;
    printf("Enter 2 numbers : ");
    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);
}

```

Expected Output:

Enter 2 numbers : 50 25

1. Addition 2. Subtraction 3. Multiplication 4. Division:

Enter your choice number : 4

Result = 2.000000

Q 5. Differentiate while & do-while loop. Write a C program to find square root of a number with do...while loop.

10 Marks

Ans:

- **while** is a pre-tested (entry-controlled) loop. Condition is checked before entering in the loop.
- **do-while** is a post-tested (exit-controlled) loop. Condition is checked after execution of the statements within the loop.

➤ The main difference is that do-while loop must execute once (because condition is checked at the end.)

while statement syntax:

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

/*working example – to display squares of numbers from 1 to 20 */

```
#include <stdio.h>
```

```
int main()
```

```
{
    int x;
    x=1;
    while(x<=20)
    {
        printf("%d %d\n", x, x*x);
        x++;
    }
    return (0);
}
```

do...while statement syntax: (It is a post tested loop & must executes at least once)

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

/*working example – to display squares of numbers from 1 to 20 */

```
#include <stdio.h>
```

```
int main()
```

```
{
    int x;
    x=1;
    do
    {
        printf("%d %d\n", x, x*x);
        x++;
    }while(x<=20);
    return (0);
}
```

/* finding sqrt of a number without using sqrt() built in method*/

```
#include <stdio.h>
```

```
#include <math.h>
```

```

int main()
{
    double num,x1,x2,diff;
    printf("Enter any number : ");
    scanf("%lf",&num);
    x1=num/2.00;
    x2=num/x1;
    do {
        x1 = (x1+x2)/2.00;
        x2 = num/x1;
        diff=x1-x2;
    }while(diff>0.000001);
    printf("Square root = %.3lf\n",x1);
    return (0);
}

```

Expected output:

Enter any number: 2

Square root = 1.414213

Q 6 a. Explain type casting/conversion with example code.

5 Marks

Ans:

Type conversion concept in C language is used to modify a variable from one data type to another data type. New data type should be mentioned before the variable name or value in brackets which to be typecast.

Example:

```
result = (float) 20/3;
```

- It is best practice to convert lower data type to higher data type to avoid data loss.
- Data will be truncated when higher data type is converted to lower. For example, if float is converted to int, data which is present after decimal point will be lost

Type conversion can be done 2ways:

a) Implicit Type conversion / Coercion / Automatic type conversion

It is done by compiler automatically:

example:

```
printf("%f", 20/3.00);
```

b) Explicit Type conversion / A cast

By preceding the expression with type in parenthesis

It may be checked, unchecked or bit pattern.

Syntax:

(type) expression;

Example:

```
result=(float) 20/3;
```

Q 6 b. Evaluate the expressions:

5 Marks

- i) $100\%20 \leq 20-5+100\%10-20 == 5 \geq 1 != 20$
- ii) $a += b * c -= 10$ (Initial values, a=5,b=10 and c=15)

Ans:

- i) $100\%20 \leq 20-5+100\%10-20 == 5 \geq 1 != 20$

Evaluating

$0 \leq 20-5+0-20 == 5 \geq 1 != 20$ (Arithmetic operator then relational operator)

$0 \leq -5 == 5 \geq 1 != 20$

$1 == 5 \geq 1 != 20$

$0 \geq 1 != 20$

$0 != 20$

1 Ans.

- ii) $a += b * c -= 10$ (Initial values, a=5,b=10 and c=15)

Evaluating (Associativity of assignment operator R to L)

c=10 will be 5

b*=c will be 50

a+=b will be 55

55 Ans.

Q 7. Write a C Program to display prime numbers within given range.

10 Marks

Ans:

```
/* To display prime numbers within given range */
#include <stdio.h>
#include <math.h>
int main()
{
    int n,n1,n2,d,prime;
    printf("Enter range from : ");
    scanf("%d",&n1);
    printf("to : ");
    scanf("%d",&n2);
    printf("prime numbers from %d to %d:\n",n1,n2);
    for(n=n1;n<=n2;n++)
    {
        prime=1;
        for(d=2;d<=sqrt(n);d++)
            if(n%d==0) { prime=0; break; }
        if(prime) printf(" %d",n);
    }
    return (0);
}
```

Expected Output:

Enter range from : 2

to : 100

prime numbers from 2 to 100:

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

Q 8. Write a C program to find reverse of a number & to check whether it is palindrome or not. 10 Marks

Ans:

```
/* To display reverse of a number
& to check whether it is palindrome */
#include <stdio.h>
#include <math.h>
int main()
{
    long int num,temp,rem,rev=0;
    printf("Enter number : ");
    scanf("%ld",&num);
    temp=num;
    while(temp!=0)
    {
        rem=temp%10;
        rev=rev*10+rem;
        temp/=10;
    }
    printf("Reverse of number is %ld\n",rev);
    if(rev==num)
        printf("Yes, %ld is palindrome\n",num);
    else
        printf("No, %ld is not palindrome\n",num);
    return (0);
}
```

Expected output:

Enter number : 12345

Reverse of number is 54321

No, 12345 is not palindrome

Enter number : 12321

Reverse of number is 12321

Yes, 12321 is palindrome