



**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
FACULTY OF ENGINEERING AND TECHNOLOGY
RAMAPURAM CAMPUS, CHENNAI-600 089**



**DEPARTMENT OF COMPUTER SCIENCE
AND
ENGINEERING**

PROGRAMMING AND PROBLEM SOLVING - 18CSS101J

Year: I

QUESTION BANK

**UNIT I [CLO 1][CLO 6]
PART – A
MULTIPLE CHOICE QUESTIONS**

1. Who is father of C Language?

- A. Bjarne Stroustrup
- B. Dennis Ritchie
- C. James A. Gosling
- D. Dr. E.F. Codd

Ans: B

2. C Language developed at _____?

- A. AT & T's Bell Laboratories of USA in 1972
- B. AT & T's Bell Laboratories of USA in 1970
- C. Sun Microsystems in 1973
- D. Cambridge University in 1972

Ans: A

3. For 16-bit compiler allowable range for integer constants is _____?

- A. -3.4e38 to 3.4e38
- B. -32767 to 32768
- C. -32768 to 32767
- D. -32668 to 32667

Ans: C

4. C programs are converted into machine language with the help of

- A. An Editor
- B. A compiler
- C. An operating system
- D. None of the above

Ans: B

5. A C variable can start with

- A. An alphabet
- B. A number
- C. A special symbol other than underscore
- D. keyword

Ans: A

6. Which of the following is allowed in a C Arithmetic Instruction?

- A. []
- B. {}
- C. ()
- D. ||

Ans: C

7. Which of the following is the correct hierarchy of arithmetic operations?

- A. / + * -
- B. * - / +
- C. + - / *
- D. * / + -

Ans: D

8. Program which is written originally by the programmer is classified as

- A. object code
- B. machine code
- C. source program
- D. interactive programs

Ans: C

9. Which operator has the lowest priority?

- A. ++
- B. %
- C. +
- D. ||

Ans: D

10. Loop statement which is repeated for some given number of times is classified as _____

- A. FOR loop
- B. GO loop
- C. REPEAT loop
- D. GO REPEAT loop

Ans: A

11. Type of statement written in sequence and is repeated until the specific condition met is classified as

- A. format
- B. loop
- C. case
- D. condition

Ans: B

12. Size of an array is declared by

- A. programmer
- B. program users
- C. software
- D. declared automatically

Ans: A

13. _____ comment is correct s about the ++ operator among the following

- A. It is not an unary operator
- B. The operator can come between the operands
- C. operand It can be applied to an expression
- D. It associates from the right

Ans: D

14. Functions that are used in the programs and are defined by the programmers are called

- A. program layout
- B. program procedure
- C. built-in functions
- D. user-defined function

Ans: D

15. An assembler trAnslates

- A. machine code into assembly code
- B. assembly code into machine code
- C. processing time into manual time
- D. routine into subroutine

Ans: B

16. Name given by a programmer to any particular data is classified as

- A. identifier
- B. identification
- C. exponent
- D. mantissa

Ans: A

**17. The ternary conditional expression using the operator?: could be easily coded using -----
----- statement**

- a) Nested if
- b) if-else
- c) if
- d) for

Ans: b

18. The ----- statement when executed in a switch statement causes immediate exit from the structure.

- a) goto
- b) default
- c) break
- d) switch

Ans: c

19. Variable which uses the same name in whole program and in its all routines thus best classified as

- A. middle variable
- B. default variable
- C. local variable
- D. global variable

Ans: D

20. Statement which is used to make choice between two options and only option is to be performed is written as

- A. if statement
- B. if else statement
- C. then else statement
- D. else one statement

Ans: A

21. The _____ statement is used to transfer the control to the end of statement block in a loop:

- a. Continue
- b. Break
- c. Switch
- d. Goto

Ans b

22. Which of the following is not a valid variable name declaration?

- a) int _a3;
- b) int a_3;
- c) int 3_a;
- d) int _3a

Ans: c

23. All keywords in C are in

- a) LowerCase letters
- b) UpperCase letters
- c) CamelCase letters
- d) None of the mentioned

Ans: a

24. Which of the following is true for variable names in C?

- a) They can contain alphanumeric characters as well as special characters
- b) It is not an error to declare a variable to be one of the keywords (like goto, static)
- c) Variable names cannot start with a digit
- d) Variable can be of any length

Ans: c

25. The format identifier '%i' is also used for _____ data type?

- a) char
- b) int
- c) float
- d) double

Ans: b

26. In an expression involving || operator, evaluation

- a. Will be stopped if one of its components evaluates to false
- b. Will be stopped if one of its components evaluates to true
- c. Takes place from right to left
- d. Takes place from left to right

Ans: d

27. What is the output of this C code? [PLO 3]

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

```
int main()
```

```
{
```

```
signed char chr;
```

```
chr = 128;
```

```
printf("%d\n", chr);
```

```
return 0;
```

```
}
```

a) 128

b) -128

c) Depends on the compiler

d) 1

Ans: b

28. What is the output of this C code?

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

```
int main()
```

```
{
```

```
j = 10;
```

```
printf("%d\n", j++);
```

```
return 0;
```

```
}
```

a) 10

b) 11

c) Compile time error

d) 0

Ans: c

29. The following code 'for(; ;)' represents an infinite loop. It can be terminated by

a) break

b) exit(0)

c) abort()

d) all of the mentioned

Ans: a

30. The keyword 'break' cannot be simply used within:

a) do-while

b) if-else

c) for

d) while

Ans: b

31. Which keyword is used to come out of a loop only for that iteration?

a) break

- b) continue
- c) return
- d) none of the mentioned

Ans: b

32. The first step in problem solving is -----

- a) Understand the problem
- b) Identify the problem
- c) Developing algorithm/Flowchart
- d) Listing the possible outcome

Ans: b

33. In C programming language, which of the following type of operators have the highest precedence

- a. Relational operators
- b. Logical operators
- c. Arithmetic operators
- d. Equality operators

Ans: d

34. The solution can be reached by completing the actions in steps. These steps are called

- a) Sequence
- b) Algorithm
- c) Flowchart
- d) Steps

Ans: b

38. ----- means the set of instructions that make up the solution after they have been coded into a programming language.

- a) solution
- b) result
- c) program
- d) error

Ans: c

39. A language applied for wide range of application is called as _____ language

- a. Special purpose
- b) **General purpose**
- c. Individual purpose
- d) Scientific purpose

Ans: b

40. Compiler converts the source code into _____

- a. C code
- b. Byte code
- c. Object code
- d. Executable code

Ans: c

41. _____ converts source code to machine language one line at a time.

- a. Compiler

- b. Interpreter
- c. Assembler
- d.CPU

Ans: b

42. The program that converts high level language to a machine language is called _____

- a. Interpreter
- b. Linker
- c. Compiler
- d.Loader

Ans: c

43. '#' symbol is known to be _____

- a. Linker
- b.Compiler
- c. Assembler
- d.Preprocessor directive

Ans: d

44. A _____ is a notational system for describing computations in both machine and human readable form

- a. Programming language
- b.Machine language
- c. High-level language
- d.Assembly language

Ans: a

45. ----- program converts assembly language into machine language

- a. compiler
- b. interpreter
- c. assembler
- d. preprocessor directive

Ans: c

46. ----- is a distinguishing characteristic of human excellence in every area of behavior

- a) creativity
- b) thinking
- c) visualization
- d) problem solving

Ans: a

47.Which of the following is the correct way of writing comments?

- a)*/comments/*
- b)/*comment*/
- c)**comment**
- d){comment}

Ans: b

48. C programming language is

- a)object oriented programming language
- b)Procedure oriented programming language

- c)function oriented programming language
- d)Unstructured programming language

Ans: b

49.The memory space taken for a char type data is

- a)2 bytes
- b)4 bytes
- c)8 bytes
- d)1bytes

Ans: d

50 .The memory space taken for aint type data is

- a) 2 bytes
- b) 4 bytes
- c) 8 bytes
- d)10bytes

Ans: a

51. What is the only function all programs must contain ?

- a)start()
- b)system()
- c)main()
- d)program

Ans: c

52.For 16-bit compiler allowable range for integer constants is _____?

- a) -3.4e38 to 3.4e38
- b) -32767 to 32768
- c) -32668 to 32667
- d) -32768 to 32767

Ans: d

53.Every statement in C program is to be terminated by a_____

- a)dot(.)
- b)semi-colon(;
- c)colon(:)
- d)Question mark(?)

Ans: b

54. which of the following will not valid expressions in C?

- a) a=2+(b=5);
- b) a=11%3
- c) a=b=c=5
- d) b+5=2

Ans: d

55. The ----- operator is true only when both the operands are true.

- a) &&
- b) ||
- c) !
- d) ?:

Ans: a

56. The ----- statement when executed in a switch statement causes immediate exit from the structure.

- a) goto
- b) default
- c) break
- d) switch

Ans: c

56. The ternary conditional expression using the operator?: could be easily coded using ----- statement

- a) Nested if
- b) if-else
- c) if
- d) for

Ans: b

UNIT I [CLO 1][CLO 6]

PART –B

4 MARKS

1. Comment “C is mid level language”?
2. What is problem solving?
3. What are the six steps of problem solving?
4. Discuss about how the problems can be solved with computers?
5. What is a program?
6. Why is problem analysis important?
7. What are the tools of problem solving available?
8. How do problem-solving tools help in leading to a solution?
9. Why it is important to test a solution before coding it?
10. What is an algorithm? Give the characteristics of the algorithm.
11. What is a flowchart? Give the symbols/shapes used in the flowchart.
12. Define pseudocode and give its importance with an example.
13. Discuss the difficulties with problem solving in detail.
14. State the use of %d and %f .Write a printf statement in C using the above mentioned symbols?
15. What is main difference between variable and constant?
16. Explain bitwise left shift operator?
17. Explain primary data types used in C?
18. Difference between formatted & unformatted statement ?
19. What is meant by storage class of variable?
20. Explain with example ++i and i++.
21. Explain about arrow operator in C.
22. What are the non-numeric data types in C ?
23. Draw a flow chart to check if a number is even.
24. What are the various data types that the scanf statement can handle

UNIT-1 [CLO 1]

PART -C

12 MARKS

- 1) Explain the evolution of programming languages.
- 2) Explain the various steps involved in problem solving with diagram.
- 3) Draw the flowchart and write the algorithm and c code to find the sum and to reverse the digits of given five-digit number.
- 4) Write an algorithm and draw a flow chart to find the factorial and Fibonacci series of given number.
- 5) Write a note on Algorithm, Flow chart and Pseudocode.
- 6) Explain the scope, lifetime of variable in C with example.
- 7) Write down the algorithm to find the largest number among three given numbers and outline the steps in the algorithm with the inputs 5, 17, 3.
- 8) Draw flowchart to compute the salary of an employee in a company. Assume that there are two types of employees in the company daily wages and regular. Salary is calculated as number of hours worked* wages per hour for daily wagers and basic pay +(% of DA * basic pay)/100 + HRA + medical allowance for regular employees. Sketch the flow of your design for a regular employee with basic pay = 5000, % of DA = 75 % and HRA = 500. (8)
- 9) Explain in detail about operators with an example
- 10) Refer all elab Programs
- 11) Draw a flowchart and write an algorithm and the code in C to find the sum of two numbers and return the digit in lowest places value.
- 12) Draw a flowchart and write an algorithm to find the product of two numbers without using the “*” operator.
- 13) Draw a flowchart and write an algorithm to find if a string is a palindrome.

**UNIT - II [CLO 2][CLO6]
PART – A
MULTIPLE CHOICE QUESTIONS**

1. The ----- operator is true only when both the operands are true.

- a) &&
- b) ||
- c) !
- d) ?

Ans: a

2. The ----- statement when executed in a switch statement causes immediate exit from the structure.

- a) goto
- b) default
- c) break
- d) switch

Ans: c

3. The ternary conditional expression using the operator?: could be easily coded using ----- statement

- a) Nested if
- b) if-else
- c) if
- d) for

Ans: b

4. What will be the output when the following segment is executed?

```
Char ch='a';  
Switch(ch)  
{  
case 'a':  
Printf("A");  
case 'b':  
Printf("B");  
Default:  
printf("C");  
}
```

- a) A
- b) B
- c) C
- d) a

Ans: a

5. What will be the output of the following segment when executed?

```
int x=10, y=20;  
if((x<y)|| (x+5)>10)  
printf("%d",x);  
else  
printf("%d",y);  
a) 10
```

- b) 20
 - c) 15
 - d) 5
- Ans: a**

6. The ----- statement is used to skip a part of the statements in a loop.

- a) Continue
- b) break
- c) goto
- d) switch

Ans: a

7. A for loop with no test condition is known as ----- loop

- a) Infinite
- b) time delay
- c) for
- d) Incrementing

Ans: a

8. The sentinel –controlled loop is also known as -----loop

- a) Indefinite repetition loop
- b) time delay
- c) Definite repetition loop
- d) infinite

Ans: a

9. In an exit controlled loop the body of the loop is executed at least for -----

- a) 1 time
- b) 2 times
- c) 3 times
- d) n times

Ans: a

10. The while is an ----- loop statement.

- a) Entry-controlled
- b) exit-controlled
- c) indefinite repetition
- d) definite repetition

Ans: a

11. The expression $!(x!=y)$ can be replaced by the expression-----

- a) $x!=y$
- b) $x==y$
- c) $x=!y$
- d) $!x=!y$

Ans: c

12. In a counter controlled loop, variable known as -----is used to count the loop operations.

- a) Counter
- b) sentinel
- c) i
- d) n

Ans: a

13. int a=10;
++a;
a++;
Printf("%d",a);
a)10
b) 11
c)12
d) 13

Ans: c

14. int a=11;
a=a%2;
a=a/2;
printf("%d",a);
a) 1
b) 5
c) 0
d)0.5

Ans: c

15.#include <stdio.h>
int main()
{
int i = 3;
printf("%d", (++i)++);
return 0;
}

What is the output of the above program?

a) 3
b) 4
c) 5
d) Compile-time error

Ans: d

16.Which of the following is not a logical operator?

a) &&
b) !
c) ||
d) |

Ans: d

17. What is the output of this C code?

```
int main()
{
int a = 0, i = 0;{
for (i = 0;i < 5; i++)
a++;
continue;
}
Printf("%d",a);
}
```

a) 2 b) 3 c) 4 d) 5

Ans : d

18. Give the output for below code

```
int main()
{
int a = 1;
int b = 1;
int c = a || --b;
int d = a-- && --b;
printf("a = %d, b = %d, c = %d, d = %d", a, b, c, d);
return 0;
}
```

- a) a = 0, b = 1, c = 1, d = 0
- b) a = 0, b = 0, c = 1, d = 0
- c) a = 1, b = 1, c = 1, d = 1
- d) a = 0, b = 0, c = 0, d = 0

Ans : b

19. What is the output of below code?

```
int main
{
int a = 10, b = 20, c = 30;
if (c > b > a)
printf("TRUE");
else
printf("FALSE");
return 0;
}
```

- a) TRUE
- b) FALSE
- c) Compiler Error
- d) Output is compiler dependent

Ans: b

20.int main() {
int a = 1;
int b = 0;
b = a++ + a++;
printf("%d %d",a,b);
return 0;
} Find the output of the code

- a) 3 6
- b) Compiler Dependent
- c) 3 4
- d) 3 3

Ans:b

21. What is the output of this C code?

```
void main()
{
int i = 0;
int j = 0;
for (i = 0; i < 5; i++)
```

```

{
for (j = 0; j < 4; j++)
{
if (i > 1)
continue;
printf("Hi \n");
}
}
}

```

- a. Hi is printed 9 times
- b. Hi is printed 8 times
- c. Hi is printed 7 times
- d. Hi is printed 6 times

Ans : b

22. What is the output of this C code?

```

void main()
{
int i = 0;
for (i = 0; i < 5; i++)
if (i < 4)
{
printf("Hello");
break;
}
}

```

- a) Hello is printed 5 times
- b) Hello is printed 4 times
- c) Hello
- d) Hello is printed 3 times

Ans : c

23. What is the output of this C code?

```

int main()
{
int i = -5;
int k = i %4;
printf("%d\n", k);
}

```

- a) Compile time error
- b) -1
- c) 1
- d) Run time error

Ans: b

24. What is the value of x in this C code?

```

void main()
{
int x = 4 *5 / 2 + 9;
}

```

- a) 6.75
- b) 1.85

- c) 19
- d) 3

Ans: c

```
25. #include <stdio.h>
void main()
{
    int a = 3;
    int b = ++a + a++ + --a;
    printf("Value of b is %d", b);
}
```

- a) Value of x is 12
- b) Value of x is 13
- c) Value of x is 10
- d) Undefined behavior

Ans: d

```
26. int main()
{
    printf("before continue ");
    continue;
    printf("after continue\n");
}
```

- a. Before continue after continue
- b. Before continue
- c. after continue
- d. Compile time error

Ans: d

28. What is the output of the code given below?

```
int main()
{
    printf("%d ", 1);
    goto l1;
    printf("%d ", 2);
l1: goto l2;
    printf("%d ", 3);
l2: printf("%d ", 4);
}
```

- a. 1 4
- b. Compilation error
- c. 1 2 4
- d. 1 3 4

Ans: a

29. What is the output of code given below?

```
int main()
{
    printf("%d ", 1);
l1: l2:
    printf("%d ", 2);
    printf("%d\n", 3);
}
```


}

- a. Compilation error
- b. 1 2 3
- c. 1 2
- d. 1 3

Ans : b

30. Result of a logical or relational expression in C is?

- a. True or False
- b. 0 or 1
- c. 0 if expression is false and any positive number if expression is true
- d. None of the mentioned

Ans: b

31. Ternary operator operates on how many operands

- a) 1
- b) 2
- c) 3
- d) 4

Ans: c

32. Which Operator has the lowest precedence

- a) sizeof
- b) unary
- c) assignment
- d) comma

Ans: d

33. What is meaning of following declaration ?

int arr[20];

- a) Array of size 20 that can have integer address
- b) Integer array of size 20
- c) Array of size 20
- d) None of the above

Ans b

34. In C Programming, If we need to store word "INDIA" then syntax is as below

- a) char name[6]={ 'I', 'N', 'D', 'I', 'A', '\0' }
- b) char name[6]={ "I", "N", "D", "I", "A" }
- c) char name[6]={ 'I', 'N', 'D', 'I', 'A' }
- d) char name[] ; name="INDIA"

Ans : a

35. what is the way to initialize array?

- a) int num[6] = {2,4,12,5,45,5};
- b) int n{} = {2,4,12,5,45,5};
- c) int n{6} = {2,4,12};
- d) int n(6) = {2,4,12,5,45,5};

Ans: a

36. what will be the output of the program?

- a) 3, 2, 15
- b) 2, 3, 20
- c) 2, 1, 15
- d) 1, 2, 5

Ans: a

37. The Statements in the statement block are encapsulated within a ---?

- a) ()
- b) { }
- c) []
- d) < >

Ans : B

38) An array elements are always stored in _____ memory locations

- a) sequential
- b) Random
- c) Sequential and Random
- d) None of the above

Ans : a

39) Let x be an array. which of the following operations are illegal?

- a) ++x
- b) x+1
- c) x++
- d) x*2

options:

- a) I and II
- b) I, II and III
- c) II and III
- d) I, III and IV

Ans: d

40) what is the maximum number of dimensions an array in c may have?

- a) 2
- b) 8
- c) 20
- d) theoretically no limit. the only practical limits are memory size and compilers

Ans : d

41) size of the array need not be specified, when

- a) initialized is a part of definition
- b) it's a declaration
- c) it is a formal parameter
- d) All of these

Ans : d

42) what will be the output of the program?

- a) Equal
- b) Unequal
- c) Error
- d) None of these

Ans: b

43. Dangling Else problem can arise in which of the following statements.

- a) Conditional Branching
- b) Loops
- c) Iterative
- d) None of these

Ans :a

44)Identify the erroneous Expression

- a) x=y=2,4;
- b)res=++a*5;
- c) res=/4 ;
- d)res=a++ -b*2

Ans: c

45) Which Operator cannot be used with float operands?

- a) +
- b)-
- c)%
- d)*

Ans :C

46) Which of the following statement about for loop is true?

- a) Index value is retained outside the loop
- b) Index value can be changed from within the loop
- c) Goto can be used to jump, out of the loop
- d) All of these

Ans d

47. A Statement block consists of how many statements

- a)0
- b)1
- c)n
- d)n+1

Ans c

48.If an array is declared as arr[]={1,3,5,7,9};then what is the value of sizeof(arr[3])

- a)1
- b)2
- c)3
- d)8

Ans b

49.If an Array is declared as int arr[50],how many elements can it hold?

- a)49
- b)50
- c)51
- d)0

Ans b

50.If an array is Declared as double arr[50],how many bytes will be allocated to it

- a)50
- b)100
- c)200
- d)400

Ans c

UNIT - II [CLO 2][CLO6]

PART -B

4 MARKS

1. Give the general syntax of conditional operator?
2. Define operator in C. What role an operator plays in C program?
3. Differentiate between relational and logical operators used in C?
4. Give the types of decision making statements
5. What is the general form of switch statement?
6. Differentiate do... while.. and while... loop.
7. Determine the output of the following and justify your **Answer**

```
int main()
{
int ch;    printf("enter a value between 1 to 2:"); scanf("%d", &ch);
switch (ch, ch -1){
case 1: printf("1\n"); break; case 2: printf("2"); break;
} return 0; }
```

8. Explain the limitations of arrays.
9. Give the steps of looping process.
10. Brief the use of continue, goto, break statement and give the syntax.
11. What are loop control structures? Explain for loop, while loop and do-while loop with their syntax.
11. What do you mean by infinite loop? Give suitable of any infinite loop in a C program.
12. Explain break and continue statements with examples.
13. Determine the output of the following program justify your **Answer** in a few words.

```
int main()
{
float f = 1.1.0;
switch(f){
case 1.0: printf("one"); break;
case 2.0: printf("two"); break; default:
printf("%f", f);
}}
```

14. Write a program to print series of number divisible by 3 from 1 to 100 using for loop.
15. Write a C program to print 1, 2, 4, 8, 16, 32, 64N using do....while loop (Read N from user).
16. What are Arrays in C programming? Give the importance of Array in C language
17. What are the rules to declare one dimensional array?
18. What do you mean by compile time initialization? Give suitable example of Compile time initialization of C Array.
19. Describe the array index out of bound error in context of C array program.
20. Write a C program to calculate median of an array in C.
21. Explain about pre/post increment operator.give examples
22. Write a C program using logical operators
23. Explain about conditional and unconditional branching using FOR loop

24. Write a C program for Lvalue and R value in expression
25. Explain about Operator Precedence
26. Explain switch statement and give an example
27. Illustrate how the elements are accessed in one dimensional array.give an example
28. Explain about else if and nested if with an example
29. Discuss about assignment operator with an example

UNIT - II [CLO 2][CLO6]
PART C
12 MARKS

1. Lucy is celebrating her 15th birthday. Her father promised her that he will buy her a new computer on her birthday if she solves the question asked by him. He asks Lucy to find whether the year on which she had born is leap year or not. Help her to solve this puzzle so that she celebrates her
2. birthday happily. If her birth year is 2016 and it is a leap year display 2016 is a leap year.? Else display 2016 is not a leap year and check with other leap year conditions
3. Write a C program to print the multiplication table of an integer n upto m rows using a while loop

sample input and output

INPUT

5

4

OUTPUT

1*5=5

2*5=10

3*5=15

4*5=20

4. Write a program to generate the following pattern

```
@ @ @ @ @
  @ @ @
```

5. Write a program that determines a student's grade. The program will read three scores and determine the grade based on the following rules: score =90% =>grade=A , score >= 70% and <90% => grade=B, score>=50% and <70% =>grade=C, score<50% =>grade=F
6. i)Why switch case is better than else..if... ladder. Justify.
 ii)Write a program to read a value (1-7) and print the equivalent day of the week (i.e. 1-SUN, 2-MON.....)
 iii)Explain briefly about for loop and Nested for loop with suitable example
 iv)Write a program to print the following pattern.

```
12345
```

```
1234
```

```
123
```

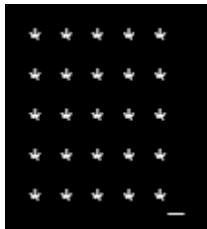
```
12
```

```
1
```

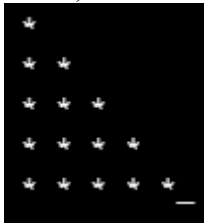
7. Write a C program to insert an element at a specified position in the array.

8. Write a program to generate a following patterns

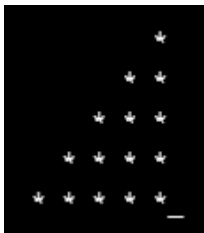
a)



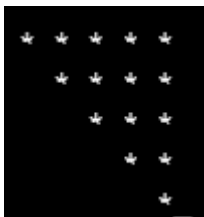
b)



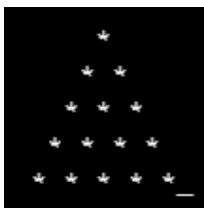
c)



d)



e)



8) Write c Program for Student mark list generation by using control and looping statements

9) Write c program for digit rotation. "For any positive integer, we define a digit rotation as either moving the first digit to the end of the number (left digit rotation), or the last digit to the front of the number (right digit rotation). For example, the number 12345 could be left digit rotated to 23451, or right digit rotated to 51234.

- 10) Write a program to find the sum of positive numbers in an array
- 11) Write c program for palindrome strings , two strings A and B, each consisting of lower case alphabets.
- 12) Write a arithmetic operation menu driven program using while(1)
- 13) Write short notes on switch case.

- 14) i)write a C program to read 10 nos. and reverse it using array
ii)Concatenate two arrays of length minimum 5 numbers.

UNIT-3 [CLO-3][CLO6]
PART – A
MULTIPLE CHOICE QUESTIONS

1. Which of the following is not possible statically in C?

- a) Jagged Array
- b) Rectangular Array
- c) Cuboidal Array
- d) Multidimensional Array

Ans- a

2. What is the output of this C code?

```
#include <stdio.h>
void main()
{
int a[2][3] = {1, 2, 3, 4, 5};
int i = 0, j = 0;
for (i = 0; i < 2; i++)
for (j = 0; j < 3; j++)
printf("%d", a[i][j]);
}
```

- a) 1 2 3 4 5 0
- b) 1 2 3 4 5 junk
- c) 1 2 3 4 5 5
- d) Run time error

Ans-a

3. Predict the output of below program:

```
#include <stdio.h>
int main()
{
intarr[5]; // Assume that base address of arr is 2000 and size of integer // is 32 bit
arr++;
printf("%u", arr);
return 0;
}
```

- a) 2002
- b) 2004
- c) 2020
- d) lvalue required

Ans-d

4. Predict the output of below program:

```
#include <stdio.h>
int main()
{
Int arr[5];
// Assume base address of arr is 2000 and size of integer is 32 bit
```



```
printf("%u %u", arr + 1, &arr + 1);  
return 0;  
}
```

- a. 2004 2020
- b. 2004 2004
- c. 2004 Garbage Value
- d. The program fails to compile because address of operator cannot be used with array name

Ans-a

5. Size of the array need not be specified, when

- a. Initialization is a part of definition
- b. It is a declaration
- c. It is a formal parameter
- d. All of these

Ans-a

6. While passing an array as an actual argument, the function call must have the array name

- a. with empty brackets
- b. with its size
- c. alone
- d. none of the above

Ans-c

7. The parameter passing mechanism for an array is

- a. call by value
- b. call by value-result
- c. call by reference
- d. none of these

Ans-c

8. Under which of the following conditions, the size of an one-dimensional array need not be specified

- a. when initialization is a part of definition
- b. when it is a declaration
- c. when it is a formal parameter and an actual argument
- d. All of the above.

Ans-d

9. If a two dimensional array is used as a formal parameter, then

- a. both the subscripts may be left empty
- b. the first (row) subscript may be left empty
- c. the first subscript must be left empty
- d. both the subscripts must be left empty

Ans-b

10. Choose the statement that best defines an array

- a. It is a collection of items that share a common name
- b. It is a collection of items that share a common name and occupy consecutive memory location
- c. It is a collection of items of the same type and storage class that share a common name and occupy consecutive memory locations
- d. None of the above

Ans-c

11. Choose the correct statements

- a. Array stores data of the same type
- b. Array can be a part of a structure
- c. Array of structure is allowed
- d. All of the above

Ans-d

12. What is the output of this C code?

```
#include <stdio.h>
void main()
{
    int a[2][3] = {1, 2, 3, 4, 5};
    int i = 0, j = 0;
    for (i = 0; i < 2; i++)
    for (j = 0; j < 3; j++)
    printf("%d", a[i][j]);
}
```

- a) 1 2 3 4 5 0
- b) 1 2 3 4 5 junk
- c) 1 2 3 4 5 5
- d) Run time error

Ans: a

13. What is the output of this C code?

```
#include <stdio.h>
void main()
{
    int a[2][3] = {1, 2, 3, , 4, 5};
    int i = 0, j = 0;
    for (i = 0; i < 2; i++)
    for (j = 0; j < 3; j++)
    printf("%d", a[i][j]);
}
```

- a. 1 2 3 junk 4 5
- b. Compile time error
- c. 1 2 3 0 4 5
- d. 1 2 3 3 4 5

Ans: b

14. What is the output of this C code?

```
1. #include <stdio.h>
2. void f(int a[][3])
3. {
4.     a[0][1] = 3;
5.     int i = 0, j = 0;
6.     for (i = 0; i < 2; i++)
7.     for (j = 0; j < 3; j++)
8.     printf("%d", a[i][j]);
9. }
10. void main()
11. {
12.     int a[2][3] = {0};
13.     fa);
```

14. }
 - a) 0 3 0 0 0 0
 - b) Junk 3 junk junk junk junk
 - c) Compile time error
 - d) All junk values

Ans- a

15. What is the output of this C code?

1. #include <stdio.h>
2. void f(int a[][])
3. {
4. a[0][1] = 3;
5. int i = 0, j = 0;
6. for (i = 0; i < 2; i++)
7. for (j = 0; j < 3; j++)
8. printf("%d", a[i][j]);
9. }
10. void main()
11. {
12. int a[2][3] = {0};
13. fa);
14. }

- a) 0 3 0 0 0 0
- b) Junk 3 junk junk junk junk
- c) Compile time error
- d) All junk values

Ans: c

16. Comment on the following statement:

int (*a)[7];

- a) An array "a" of pointers.
- b) A pointer "a" to an array.
- c) A ragged array.
- d) a pointer "a" to a value

Ans: b

17. Comment on the 2 arrays regarding P and Q:

1. int *a1[8];
2. int *(a3[8]);
3. P. Array of pointers
4. Q. Pointer to an array

a) a1 is P, a2 is Q b) a1 is P, a2 is P c) a1 is Q, a2 is P d) a1 is Q, a2 is Q

Ans: b

18. Which of the following is not possible statically in C?

- a) Jagged Array
- b) Rectangular Array
- c) Cuboidal Array

d) Multidimensional Array

Ans: a

19. What is the correct syntax to send a 3-dimensional array as a parameter? (Assuming declaration `int a[5][4][3];`)

- a) `funca;`
- b) `func(&a);`
- c) `func(*a);`
- d) `func(**a);`

Ans: a

20. Applications of multidimensional array are?

- a) Matrix-Multiplication
- b) Minimum Spanning Tree
- c) Finding connectivity between nodes
- d) All of the mentioned

Ans: d

21. What is the output of this C code?

```
1. #include <stdio.h>
2. int main()
3. {
4.     int ary[2][3];
5.     foo(ary);
6. }
7. void foo(int *ary[])
8. {
9.     int i = 10, j = 2, k;
10.    ary[0] = &i;
11.    ary[1] = &j;
12.    *ary[0] = 2;
13.    for (k = 0; k < 2; k++)
14.        printf("%d\n", *ary[k]);
15. }
```

- a) 2 2
- b) Compile time error
- c) Undefined behaviour
- d) 10 2

Ans: a

22. What is the output of this C code?

```
1. #include <stdio.h>
2. int main()
3. {
4.     int ary[2][3];
5.     foo(ary);
6. }
```

```

7. void foo(int (*ary)[3])
8. {
9. int i = 10, j = 2, k;
10. ary[0] = &i;
11. ary[1] = &j;
12. for (k = 0; k < 2; k++)
13. printf("%d\n", *ary[k]);
14. }
a) Compile time error
b) 10 2
c) Undefined behavior
d) segmentation fault/code crash

```

Ans: a

23. What is the output of this C code?

```

1. #include <stdio.h>
2. int main()
3. {
4. foo(ary);
5. }
6. void foo(int **ary)
7. {
8. int i = 10, k = 10, j = 2;
9. int *ary[2];
10. ary[0] = &i;
11. ary[1] = &j;
12. printf("%d\n", ary[0][1]);
13. }

```

- a) 10
- b) 2
- c) Compile time error
- d) Undefined behavior

Ans: d

24. What is the output of this C code?

```

1. #include <stdio.h>
2. int main()
3. {
4. int ary[2][3][4], j = 20;
5. ary[0][0] = &j;
6. printf("%d\n", *ary[0][0]);
7. }

```

- a) Compile time error
- b) 20
- c) Address of j
- d) Undefined behaviour

Ans: a

25. What is the output of this C code?

```
1. #include <stdio.h>
2. int main()
3. {
4. int ary[2][3];
5. ary[][] = {{1, 2, 3}, {4, 5, 6}};
6. printf("%d\n", ary[1][0]);
7. }
```

a) Compile time error b) 4 c) 1 d) 2

Ans: a

26. What is the output of this C code?

```
1. #include <stdio.h>
2. int main()
3. {
4. void foo();
5. printf("1 ");
6. foo();
7. }
8. void foo()
9. {
10. printf("2 ");
11. }
```

a) 1 2
b) Compile time error
c) 1 2 1 2
d) Depends on the compiler

Ans: a

27. What is the output of this C code?

```
1. #include <stdio.h>
2. int main()
3. {
4. void foo(), f();
5. f();
6. }
7. void foo()
8. {
9. printf("2 ");
10. }
11. void f()
12. {
13. printf("1 ");
```

14. foo();
15. }

- a) Compile time error as foo is local to main
- b) 1 2
- c) 2 1
- d) Compile time error due to declaration of functions inside main

Ans: b

28. What is the output of this C code?

```
1. #include <stdio.h>
2. int main()
3. {
4. void foo();
5. void f()
6. {
7. foo();
8. }
9. f();
10. }
11. void foo()
12. {
13. printf("2 ");
14. }
```

- a) 2 2
- b) 2
- c) Compile time error
- d) Depends on the compiler

Ans: d

29. What is the output of this C code?

```
1. #include <stdio.h>
2. void foo();
3. int main()
4. {
5. void foo();
6. foo();
7. return 0;
8. }
9. void foo()
10. {
11. printf("2 ");
12. }
```

- a) Compile time error

- b) 2
- c) Depends on the compiler
- d) Depends on the standard

Ans: b

30. What is the output of this C code?

```
1. #include <stdio.h>
2. void foo();
3. int main()
4. {
5. void foo(int);
6. foo(1);
7. return 0;
8. }
9. void foo(int i)
10. {
11. printf("2 ");
12. }
```

- a) 2
- b) Compile time error
- c) Depends on the compiler
- d) Depends on the standard

Ans: a

31. What is the output of this C code?

```
1. #include <stdio.h>
2. void foo();
3. int main()
4. {
5. void foo(int);
6. foo();
7. return 0;
8. }
9. void foo()
10. {
11. printf("2 ");
12. }
```

- a) 2
- b) Compile time error
- c) Depends on the compiler
- d) Depends on the standard

Ans: b

32. What is the output of this C code?

```
1. include <stdio.h>
2. void m()
3. {
4. printf("hi");
```



```
5. }  
6. void main()  
7. {  
8. m();  
9. }
```

- a) hi
- b) Run time error
- c) Nothing
- d) Varies

Ans: a

33. What is the output of this C code?

```
1. #include <stdio.h>  
2. void m();  
3. void n()  
4. {  
5. m();  
6. }  
7. void main()  
8. {  
9. void m()  
10. {  
11. printf("hi");  
12. }  
13. }
```

- a) hi
- b) Compile time error
- c) Nothing
- d) Varies

Ans: b

34. What is the return-type of the function sqrt()

- a) int
- b) float
- c) double
- d) depends on the data type of the parameter

Ans – c

35. What is the output of this code having void return-type function? (CLO4,PLO1)

```
1. #include <stdio.h>  
2. void foo()  
3. {  
4. return 1;  
5. }  
6. void main()  
7. {
```

```
8. int x = 0;
9. x = foo();
10. printf("%d", x);
11. }
```

- a) 1
- b) 0
- c) Runtime error
- d) Compile time error

Ans: d

36. What will be the data type returned for the following function?

```
1. #include <stdio.h>
2. int func()
3. {
4. return (double)(char)5.0;
5. }
```

- a) char
- b) int
- c) double
- d) multiple type-casting in return is illegal

Ans: b

37. What is the problem in the following declarations?

```
int func(int);
double func(int);
int func(float);
```

- a) A function with same name cannot have different signatures
- b) A function with same name cannot have different return types
- c) A function with same name cannot have different number of parameters
- d) All of the mentioned

Ans: d

38. The output of the code below is

```
1. #include <stdio.h>
2. void main()
3. {
4. int k = m();
5. printf("%d", k);
6. }
7. void m()
8. {
9. printf("hello");
10. }
```

- a) hello 5
- b) Error
- c) Nothing
- d) Junk value

Ans: a

39. The output of the code below is

```
1. #include <stdio.h>
2. int *m()
3. {
4. int *p = 5;
5. return p;
6. }
7. void main()
8. {
9. int *k = m();
10. printf("%d", k);
11. }
```

- a) 5
- b) Junk value c) 0 d) Error

Ans: a

40. The output of the code below is

```
1. #include <stdio.h>
2. int *m();
3. void main()
4. {
5. int *k = m();
6. printf("hello ");
7. printf("%d", k[0]);
8. }
9. int *m()
10. {
11. int a[2] = {5, 8};
12. return a;
13. }
```

- a) hello 5 8 b) hello 5 c) hello followed by garbage value d) Compilation error

Ans: c

41. The Function that is invoked is known as

- a. Calling function
- b. caller function
- c. called function
- d. invoking function

Ans- c

42. Function declaration statement identifies a function with its

- a. Name
- b. arguments
- c. data type of return value
- d. all of above

Ans- b

43. Which return type cannot return any value to the caller?

- a. int b. float c. void d. double

Ans - c

44. Memory allocated for a function when the function is,

- a. declared
- b. defined
- c. called
- d. returned

Ans – b

45. If an array is declared as arr[]={1,3,5,7,9} then what is the value of sizeof(arr[3])?

- a. 1
- b. 2
- c. 3
- d. 8

Ans - c

46. If an array is declared as arr[]={1,3,5,7,9} then what is the value of arr[3]?

- a. 1
- b. 7
- c. 9
- d. 5

Ans - c

47. If an array is declared as double arr[50] how many bytes will be allocated to it?

- a. 50
- b. 100
- c. 200
- d. 400

Ans - d

48. If an array is declared as int arr[5][5] how many elements can it store?

- a. 5
- b. 25
- c. 10
- d. 0

Ans – b

49. strcat() is defined in which of the header files?

- a. ctype.h
- b. stdio.h
- c. string.h
- d. math.h

Ans – c

50. A string can be read using which function?

- a. gets()

- b. scanf()
- c. getchar()
- d. all of these

Ans-d

51. strlen("oxford university press") is ?

- a. 22
- b.
- c.
- d. 23
- e. 24
- f. 25

Ans – b

52. Which function adds a string to the end of another string?

- a. stradd()
- b. strcat()
- c. strok()
- d. strcpy()

Ans - b

UNIT-3 [CLO-3][CLO6]
PART B

4 MARKS

1. What is the need for user defined function?
2. Mention the types of functions with example C program.
3. List the Elements of user defined function.
4. Give short note on 2-D array processing.
5. Write a C program to find the length of a string.
6. Write a program in C to find the sum of all elements of the array.
7. Show the difference between actual and formal parameter in function.
8. Compare User define function with System define function.
9. List out the categories of functions with example program.
10. Explain the concept of function call by reference with a example program.
11. Explain the concept of function call by value with a sample program.
12. How a function is defined? Explain with relevant example.
13. Discuss the character of Arrays and types with example program.
14. Predict output of the following program.

```
int main(){
inta[][][]={{1,2},{3,4},{5,6}};
int i,j;

for(j=0;j<2;j++)
printf("%d",a[i][j]);
return 0; }
```
15. Why array index starts from zero? Explain?.
16. Differentiate function declaration and function definition..
17. List the advantages of functions. .
18. Write a program to find the 2nd Largest Elements in an Array. .
19. Write a program to convert the given string (computer) Lower to upper case.
20. State the limitations of multi dimensional array initialization. .
21. Explain call by value and call by reference with an example..
22. Discuss the common programming errors in 2D arrays.
23. Write a program based on array to find out the average of 4 integers..
24. Write a C program to concatenate two strings.
25. How single dimensional array is defined?.

UNIT-3 [CLO-3][CLO6]

PART C

12 MARKS

1. Array construction for student mark list for 25 students. Output need to display register number, marks of five subjects , CGPA and PASS/Fail Status.
2. Explain in details about String Functions: gets(), puts(), getchar(), putchar(), printf() with an example programs.
3. i)Print the given pattern using 1-D array.

4789456

123544

56478

123

- ii)Write a C program to divide one array into two arrays.
4. i) What are strings in C? Write a C program to read a string in lowercase and convert it to uppercase.
- ii) Illustrate call by value and call by reference with example for each..
5. i) Find the factorial of 10 using function recursion.
- ii) What is a string? Explain any 5 string functions..
6. Explain in details about String Functions: atoi, strlen, strcat, strcmp with an example.
7. Discuss about functions declaration and definition, Types: Call by Value & Call by Reference with example programs..
8. Explain and write c program for Function with and without Arguments and no Return Values..
9. Explain and write c program for Passing Array to Functions with return type, Recursion Functions..
10. Write c program for Matrix Multiplication using Multi-dimensional array.
11. Write a program to perform Matrix addition and Multiplication using 2-D arrays.
12. State the importance of functions. List out the different types of Function.
13. Write c program for Matrix addition and Matrix Transpose using multi dimensional array.
14. Given a number , find whether it is a power of 2 or not ?
15. Write a C program to swap elements in cyclic order using call by reference..
16. Write a C program that: i. Implements string copy operation STRCOPY(str1,str2) that copies a string str1 to another string str2 without using library function. ii. Reads a sentence and prints frequency of each of the vowels and total count of consonants..
- 17.Explain any five string manipulation library functions with examples.
- 18.Write a program in C to merge two arrays of same size sorted in decending order.
19. Write a swap function using call by value and call by reference. 20. Write a program to read to strings and compare them and print a message that the first string is equal, less or greater than the second one accordingly.

UNIT-4 [CLO 4][CLO6]
PART-A
MULTIPLE CHOICE QUESTIONS

1. In C, parameters are always

- a) Passed by value
- b) Passed by reference
- c) Non-pointer variables are passed by value and pointers are passed by reference
- d) Passed by value result

Ans: a)

2. Which of the following is true about return type of functions in C?

- a) Functions can return any type
- b) Functions can return any type except array and functions
- c) Functions can return any type except array, functions and union
- d) Functions can return any type except array, functions, function pointer and union

Ans: b)

3. #include <stdio.h>

```
int main()
{
    printf("%d", main);
    return 0;
}
```

- a) Address of main function
- b) Compiler Error
- c) Runtime Error
- d) Some random value

Ans: a)

4. Output?

```
filter_none
edit
play_arrow
brightness_4
#include <stdio.h>
int main()
{
    int (*ptr)(int ) = fun;
    (*ptr)(3);
    return 0;
}
int fun(int n)
{
    for(;n > 0; n--)
        printf("GeeksQuiz ");
    return 0;
}
```

- a) GeeksQuiz GeeksQuiz GeeksQuiz
- b) GeeksQuiz GeeksQuiz

- c) Compiler Error
- d) Runtime Error

Ans: c)

5. What is the meaning of using extern before function declaration?

For example following function sum is made extern

```
extern int sum(int x, int y, int z)
```

```
{  
    return (x + y + z);  
}
```

- a) Function is made globally available
- b) extern me**Ans** nothing, sum() is same without extern keyword.
- c) Function need not to be declared before its use
- d) Function is made local to the file.

Ans: b)

6. What is the meaning of using static before function declaration?

For example following function sum is made static

```
static int sum(int x, int y, int z)
```

```
{  
    return (x + y + z);  
}
```

- a) Static me**Ans** nothing, sum() is same without static keyword.
- b) Function need not to be declared before its use
- c) Access to static functions is restricted to the file where they are declared
- d) Static functions are made inline

Ans: c)

7. The default parameter passing mechanism is

- a) call by value
- b) call by reference
- c) call by value result
- d) None of these

Ans: a)

8. Which of the following is a complete function?

- a) int funct();
- b) int funct(int x) { return x=x+1; }
- (c) void funct(int) { printf("Hello"); }
- d) void funct(x) { printf("Hello"); }

Ans: b

9. The recursive functions are executed in a

- a) Parallel order
- b) First In First Out order
- (c) Last In First Out order
- d) Iterative order

Ans: (c)

10. What will be the output of the following C code?

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

```

void main()
{
    int k = 5;
    int *p = &k;
    int **m = &p;
    printf("%d%d%d\n", k, *p, **m);
}

```

- a) 5 5 5
- b) 5 5 junk value
- c) 5 junk junk
- d) Run time error

Ans: a

11. What will be the output of the following C code?

```

#include <stdio.h>
void main()
{
    int k = 5;
    int *p = &k;
    int **m = &p;
    printf("%d%d%d\n", k, *p, **p);
}

```

- a) 5 5 5
- b) 5 5 junk value
- c) 5 junk junk
- d) Compile time error

Ans: d

12. What will be the output of the following C code?

```

#include <stdio.h>
void main()
{
    int k = 5;
    int *p = &k;
    int **m = &p;
    **m = 6;
    printf("%d\n", k);
}

```

- a) 5
- b) Compile time error
- c) 6
- d) Junk

Ans: c

13. What will be the output of the following C code?

```

#include <stdio.h>
void main()
{
    int a[3] = {1, 2, 3};
    int *p = a;
    int *r = &p;
    printf("%d", (**r));
}

```

- a) 1
- b) Compile time error
- c) Address of a
- d) Junk value

Ans: b

14. What will be the output of the following C code?

```
#include <stdio.h>
void main()
{
    int a[3] = {1, 2, 3};
    int *p = a;
    int **r = &p;
    printf("%p %p", *r, a);
}
```

- a) Different address is printed
- b) 1 2
- c) Same address is printed
- d) 1 1

Ans: c

15. How many number of pointer (*) does C have against a pointer variable declaration?

- a) 7
- b) 127
- c) 255
- d) No limits

Ans: d

16. What will be the output of the following C code?

```
#include <stdio.h>
int main()
{
    int a = 1, b = 2, c = 3;
    int *ptr1 = &a, *ptr2 = &b, *ptr3 = &c;
    int **sptr = &ptr1; //Ref
    *sptr = ptr2;
}
```

- a) ptr1 points to a
- b) ptr1 points to b
- c) sptr points to ptr2
- d) none of the mentioned

Ans: b

17. What will be the output of the following C code?

```
#include <stdio.h>
void main()
{
    int a[3] = {1, 2, 3};
    int *p = a;
    int **r = &p;
    printf("%p %p", *r, a);
}
```

- a) Different address is printed
- b) 1 2
- c) Same address is printed
- d) 1 1

Ans: c

18. What will be the output of the following C code?

```
#include <stdio.h>
int main()
{
    char *p = NULL;
    char *q = 0;
    if (p)
        printf(" p ");
    else
        printf("nullp");
    if (q)
        printf("q\n");
    else
        printf(" nullq\n");
}
```

- a) nullp nullq
- b) Depends on the compiler
- c) x nullq where x can be p or nullp depending on the value of NULL
- d) p q

Ans: a

19. What will be the output of the following C code?

```
#include <stdio.h>
int main()
{
    int i = 10;
    void *p = &i;
    printf("%d\n", (int)*p);
    return 0;
}
```

- a) Compile time error
- b) Segmentation fault/runtime crash
- c) 10
- d) Undefined behavior

Ans: a

20. What will be the output of the following C code?

```
#include <stdio.h>
int main()
{
    int i = 10;
    void *p = &i;
    printf("%f\n", *(float*)p);
    return 0;
}
```

- a) Compile time error
- b) Undefined behaviour

- c) 10
- d) 0.000000

Ans: d

21. What will be the output of the following C code?

```
#include <stdio.h>
int *f();
int main()
{
    int *p = f();
    printf("%d\n", *p);
}
int *f()
{
    int *j = (int*)malloc(sizeof(int));
    *j = 10;
    return j;
}
a) 10
```

- b) Compile time error
- c) Segmentation fault/runtime crash since pointer to local variable is returned
- d) Undefined behaviour

Ans: a

22. What will be the output of the following C code?

```
#include <stdio.h>
void foo(int*);
int main()
{
    int i = 10;
    foo(&i++);
}
void foo(int *p)
{
    printf("%d\n", *p);
}
a) 10
```

- b) Some garbage value
- c) Compile time error
- d) Segmentation fault/code crash

Ans: c

23. What will be the output of the following C code?

```
#include <stdio.h>
void foo(int*);
int main()
{
    int i = 10, *p = &i;
    foo(p++);
}
void foo(int *p)
{
    printf("%d\n", *p);
}
```

```
}
```

a)10

b) Some garbage value

c) Compile time error

d) Segmentation fault

Ans: a

24. What will be the output of the following C code?

```
#include <stdio.h>
void foo(float *);
int main()
{
    int i = 10, *p = &i;
    foo(&i);
}
void foo(float *p)
{
    printf("%f\n", *p);
}
```

a)10.000000

b) 0.000000

c) Compile time error

d) Undefined behaviour

Ans: b

25. What will be the output of the following C code?

```
#include <stdio.h>
int main()
{
    int i = 97, *p = &i;
    foo(&i);
    printf("%d ", *p);
}
void foo(int *p)
{
    int j = 2;
    p = &j;
    printf("%d ", *p);
}
```

a)2 97

b) 2 2

c) Compile time error

d) Segmentation fault/code crash

Ans: a

26. What will be the output of the following C code?

```
#include <stdio.h>
int main()
{
    int i = 97, *p = &i;
    foo(&p);
    printf("%d ", *p);
    return 0;
}
```

```

}
void foo(int **p)
{
    int j = 2;
    *p = &j;
    printf("%d ", **p);
}

```

- a) 2 2
- b) 2 97
- c) Undefined behaviour
- d) Segmentation fault/code crash

Ans: a

27. What will be the output of the following C code?

```

#include <stdio.h>
int main()
{
    int i = 11;
    int *p = &i;
    foo(&p);
    printf("%d ", *p);
}
void foo(int *const *p)
{
    int j = 10;
    *p = &j;
    printf("%d ", **p);
}

```

- a) Compile time error
- b) 10 10
- c) Undefined behaviour
- d) 10 11

Ans: a

28. What will be the output of the following C code?

```

#include <stdio.h>
int main()
{
    int i = 10;
    int *p = &i;
    foo(&p);
    printf("%d ", *p);
    printf("%d ", *p);
}
void foo(int **const p)
{
    int j = 11;
    *p = &j;
    printf("%d ", **p);
}

```

- a) 11 11 11
- b) 11 11 Undefined-value
- c) Compile time error

d) Segmentation fault/code-crash

Ans: b

29. What will be the output of the following C code?

```
#include <stdio.h>
int main()
{
    int i = 10;
    int *const p = &i;
    foo(&p);
    printf("%d\n", *p);
}
void foo(int **p)
{
    int j = 11;
    *p = &j;
    printf("%d\n", **p);
}
```

a) 11 11

b) Undefined behaviour

c) Compile time error

d) Segmentation fault/code-crash

Ans: a

30. Which of the following is the correct syntax to send an array as a parameter to function?

a) func(&array);

b) func(#array);

c) func(*array);

d) func(array[size]);

Ans: a

31. What will be the output?

```
main()
{
    char *p;
    p = "Hello";
    printf("%cn", *&*p);
}
```

a) Hello

b) H

c) Some address will be printed

d) None of these

Ans: B

32. Determine output:

```
#include <stdio.h>
void main()
{
    char *p = NULL;
    char *q = 0;
    if(p)
```



```

    printf(" p ");
else
    printf("nullp");
if(q)
    printf("q");
else
    printf(" nullq");
}

```

- a) p q
- b) Depends on the compiler
- c) x nullq where x can be p or nullp depending on the value of NULL
- d) nullp nullq

Ans: D

33. The address operator &, cannot act on

- a) R-values
- b) Arithmetic expressions
- c) Both of the above
- d) Local variables

Ans: C

34. The statement `int **a;`

- a) is illegal
- b) is legal but meaningless
- c) is syntactically and semantically correct
- d) None of these

Ans: C

35. The operator > and < are meaningful when used with pointers, if

- a) The pointers point to data of similar type
- b) The pointers point to structure of similar data type.
- c) The pointers point to elements of the same array
- d) None of these

Ans: C

36. The declaration

`int (*p) [5];` **meAns**

- a) p is one dimensional array of size 5, of pointers to integers
- b) p is a pointer to a 5 elements integer array
- c) The same as `int *p[`
- d) None of these.

Ans: B

37. Comment on the following?

`const int *ptr;`

- a) We cannot change the value pointed by ptr
- b) We cannot change the pointer ptr itself
- c) Both of the above
- d) We can change the pointer as well as the value pointed by it.

Ans: A

38. A function 'p' that accepts a pointer to a character as argument and returns a pointer to an

array of integer can be declared as

- a) `int(*p(char *))[]`
- b) `int *p(char *)[]`
- c) `int (*p) (char *)[]`
- d) None of these

Ans: A

39.What will be printed after compiling and running the following code?

```
main()
{
    char *p;
    printf("%d %d",sizeof(*p), sizeof(p));
}
```

- a) 1 1
- b) 1 2
- c) 2 1
- d) 2 2

Ans: B

40. What is (void*)0?

- A. Representation of NULL pointer
- B. Representation of void pointer
- C. Error
- D. None of above

Ans: Option A

41. Can you combine the following two statements into one?

```
char *p;
p = (char*) malloc(100);
```

- a. `char p = *malloc(100);`
- b. `char *p = (char) malloc(100);`
- c. `char *p = (char*)malloc(100);`
- d. `char *p = (char *)(malloc*)(100);`

Ans: c

42.How many bytes are occupied by near, far and huge pointers (DOS)?

- a. near=2 far=4 huge=4
- b. near=4 far=8 huge=8
- c. near=2 far=4 huge=8
- d. near=4 far=4 huge=8

Ans: a

PART-B
4 MARKS

- 1) Write short notes on Passing Array Element to function with an example.
- 2) Explain Formal and Actual Parameters with examples.
- 3) Differentiate pass by value and pass by reference.
- 3) Give the advantages of Functions.
- 4) Define Processor Directives and #define Directives.
- 5) Write a program for Pointer Declaration and dereferencing.
- 6) Write a program for Void Pointers and size of Void Pointers.
- 7) State the limitations of pointers.
- 8) Define a recursive function with an example program.
- 9) Write a program in C to demonstrate the use of & (address of) and *(value at address) operator.
- 10) Define null pointer and void pointer.
- 11) Identify the use of null pointer.
- 12) Can pointers subtracted from each other? Justify the **Ans.**
- 13) What does it mean when a pointer is used in an "if statement"?
- 14) Find the output of the following programs

i)

```
#include<stdio.h>
int main()
{
    int arr[] = {10, 20, 30, 40, 50, 60};
    int *ptr1 = arr;
    int *ptr2 = arr + 5;
    printf("Number of elements between two pointer are: %d.",
           (ptr2 - ptr1));
    printf("Number of bytes between two pointers are: %d",
           (char*)ptr2 - (char*) ptr1);
    return 0;
}
```

- ii) Assume that float takes 4 bytes, predict the output of following program.

```
#include <stdio.h>
int main()
{
    float arr[5] = {12.5, 10.0, 13.5, 90.5, 0.5};
    float *ptr1 = &arr[0];
    float *ptr2 = ptr1 + 3;
    printf("%f", *ptr2);
    printf("%d", ptr2 - ptr1);
    return 0;
}
```

- 15) Write a program in C to swap two numbers using call by reference and call by value.
- 16) Write a program to find Sum of N Numbers using Recursion.
- 17) Mention the types of functions with illustrations.
- 18) Write the syntax for function declaration with an example program
- 19) Classify the parts of function definition.
- 20) Describe the following
 - i) Function call
 - ii) Default arguments
 - iii) Command line arguments

UNIT – 4 [CLO 4][CLO6]

PART-C

12 MARKS

- 1) Discuss about pointers and their operations that can be performed on it.
- 2) Write a c program to read array elements and print the values with their address.
- 3) Define address operator, explain with an example of Size of Pointer Variable. Explain in detail about Pointer Declaration and dereferencing pointers, Void Pointers and size of Void Pointers.
- 4) Write a program for Arithmetic Operations and Incrementing Pointers.
- 5) Write a program for Arithmetic Operations and Incrementing Pointers.
- 6) Explain function pointer & Array of Function Pointers with an example.
- 7) Write the c program for Accessing Array of Function Pointers.
- 8) Justify and explain the different size of data types using pointer variables with example program.
- 9) i) Write a c program to count vowels and consonants in a string using pointer.
ii) Explain call by reference and call by value with an example program.
- 10) Explain Constant Pointer with an example.
- 11) Explain in detail about Pass by Value and Pass by reference with an example program.
- 12) Discuss about the classification of functions depending upon their inputs and output.
- 13) Explain in detail about conditional compilation derivatives and pragma derivative.
- 14) i) Write a c- program to reverse the string using functions.
ii) Write a c- program to find the factorial of a number using functions.

UNIT 5
PART A
MULTIPLE CHOICE QUESTIONS

1. Which of the following are themselves a collection of different data types?

- a) string
- b) structures
- c) char
- d) all of the mentioned

Ans: b

2. User-defined data type can be derived by

- a) struct
- b) enum
- c) typedef
- d) all of the mentioned

Ans: d

3. Which operator connects the structure name to its member name?

- a) –
- b) <-
- c) .
- d) Both <- and .

Ans :c

4. Which of the following cannot be a structure member?

- a) Functions
- b) Array.
- c) Another structure
- d) None of the above

Ans : a

5. What is the output of this C code?

```
#include <stdio.h>
struct student
{
int no;
char name[20];
}
void main()
{
struct student s;
s.no = 8;
printf("hello");
}
```

- a) Compile time error
- b) Nothing
- c) hello

d) Varies

Ans :a

6.What is the similarity between a structure, union and enumeration?

- A. All of them let you define new values
- B. All of them let you define new data types
- C. All of them let you define new pointers
- D. All of them let you define new structures

Ans: B

9. What will be the size of the following structure?

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

```
struct temp
```

```
{  
    int a[10];  
    char p;  
};
```

- a. 5
- b. 11
- c. 41
- d. 44

Ans :d

10. Which among the following is never possible in C when members in a structure are same as that in a union?

- a. sizeof(P) is greater than sizeof(Q)
- b. sizeof(P) is equal to sizeof(Q)
- c. sizeof(P) is less than to sizeof(Q)
- d. None of the above

Ans :c

11. Predict the output of following C program

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

```
struct Point
```

```
{  
    int x, y, z;  
};
```

```
int main()
```

```
{  
    struct Point p1 = {.y = 0, .z = 1, .x = 2};  
    printf("%d %d %d", p1.x, p1.y, p1.z);  
    return 0;  
}
```

- a) Compiler Error
- b) 2 0 1

- c) 0 1 2
- d) 2 1 0

Ans: b)

12. Assume that size of an integer is 32 bit. What is the output of following program?

```
#include<stdio.h>
struct st
{
    int x;
    static int y;
};

int main()
{
    printf("%d", sizeof(struct st));
    return 0;
}
```

- a) 4
- b) 8
- c) Compiler Error
- d) Runtime Error

Ans: c)

13. Consider the following C declaration

```
struct {
    short s[5];
    union {
        float y;
        long z;
    }u;
} t;
```

Assume that objects of the type short, float and long occupy 2 bytes, 4 bytes and 8 bytes, respectively. The memory requirement for variable t, ignoring alignment considerations, is

- a) 22 bytes
- b) 14 bytes
- c) 18 bytes
- d) 10 bytes

Ans: c)

14. Output for following code.

```
#include<stdio.h>
struct st
{
    int x;
    struct st next;
};

int main()
```

```

{
    struct st temp;
    temp.x = 10;
    temp.next = temp;
    printf("%d", temp.next.x);
    return 0;
}

```

- a) Compiler Error
- b) 10
- c) Runtime Error
- d) Garbage Value

Ans: a)

15. Choose a correct statement about C structure elements.?

- a) Structure elements are stored on random free memory locations
- b) structure elements are stored in register memory locations
- c) structure elements are stored in contiguous memory locations
- d) None of the above.

Ans: c)

16. A C Structure or User defined data type is also called.?

- a) Derived data type
- b) Secondary data type
- c) Aggregate data type
- d) All the above

Ans d.

17. What are the uses of C Structures.?

- a) structure is used to implement Linked Lists, Stack and Queue data structures
- b) Structures are used in Operating System functionality like Display and Input taking.
- c) Structure are used to exchange information with peripherals of PC
- d) All the above

Ans d

18. Number of bytes in memory taken by the below structure is?

```

struct test
{
    int k;
    char c;
};

```

- a. Multiple of integer size
- b. integer size+character size
- c. Depends on the platform
- d. Multiple of word size

Ans b

19. Size of a union is determined by size of the.

- a. First member in the union
- b. Last member in the union
- c. Biggest member in the union
- d. Sum of the sizes of all members

Ans c

20. Comment on the following union declaration?

```
union temp
{
int a;
float b;
char c;
};
union temp s = {1,2.5,'A'}; //REF LINE
Which member of the union will be active after REF LINE?
```

- a. a
- b. b
- c. c
- d. Such declaration are illegal

Ans a

21. What would be the size of the following union declaration?

```
union uTemp
{
double a;
int b[10];
char c;
}u;
(Assuming size of double = 8, size of int = 4, size of char = 1)
```

- a. 4
- b. 8
- c. 40
- d. 80

Ans c

22. Which of the following structure declaration will throw an error?

- a. struct temp{}s;
main(){}
- b. struct temp{};
struct temp s;
main(){}

c. struct temp s;
struct temp{};
main(){}

d. None of the mentioned

Ans d

23. What is important difference between structure & union?

- a) There is no difference
- b) Union takes less memory
- c) Union is faster
- d) Structure is faster

Ans b

24. Which predefined function is used to write whole structure into a file ?

- a) write()
- b) fwrite()
- c) swrite()
- d) None of the above

Ans b

25.find the output of below code -

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

```
struct cppbuzz{
```

```
};
```

```
int main()
```

```
{  
    printf("%d",sizeof(struct cppbuzz));  
    return 0;  
}
```

- a) 0
- b) 1
- c) 2
- d) 4.

Ans a

26. find the output of below code on Union in C

```
#include "stdio.h"
```

```
union example{
```

```
    int a;
```

```
    int b;
```

```
};
```

```
int main(){
```

```
    union example obj;
```

```
    obj.a = 10;
```

```
    printf("%d %d", obj.a, obj.b);
```

```
    return 0;
}
```

- a) 10 0
- b) 10 GarbageValue
- c) 10 10
- d) 0 0

Ans : c

27. What is the output of C program.?

```
int main()
{ struct book
{ int pages;
char name[10];
}a;
a.pages=10;
strcpy(a.name,"Cbasics");
printf("%s=%d", a.name,a.pages);
return 0;
}
```

- a) empty string=10
- b) C=basics
- c) Cbasics=10
- d) Compiler error

Ans : c

28. What is the return type of malloc() or calloc()

- a) void *
- b) Pointer of allocated memory type
- c) void **
- d) int *

Ans : a

29. Which of the following is true?

a) “ptr = calloc(m, n)” is equivalent to following
ptr = malloc(m * n);

b) “ptr = calloc(m, n)” is equivalent to following
ptr = malloc(m * n);
memset(ptr, 0, m * n);

c) “ptr = calloc(m, n)” is equivalent to following
ptr = malloc(m);
memset(ptr, 0, m);

d) “ptr = calloc(m, n)” is equivalent to following
ptr = malloc(n);
memset(ptr, 0, n);

Ans : b

30. What is the problem with following code?

```
#include<stdio.h>
int main()
```

```

{
    int *p = (int *)malloc(sizeof(int));

    p = NULL;

    free(p);
}

```

- a) Compiler Error: free can't be applied on NULL pointer
- b) Memory Leak
- c) Dangling Pointer
- d) The program may crash as free() is called for NULL pointer.

Ans: b)

31. Consider the following program, where are i, j and k are stored in memory?

```

int i;
int main()
{
    int j;
    int *k = (int *) malloc (sizeof(int));
}

```

- a) i, j and *k are stored in stack segment
- b) i and j are stored in stack segment. *k is stored on heap.
- c) i is stored in BSS part of data segment, j is stored in stack segment. *k is stored on heap.
- d) j is stored in BSS part of data segment, i is stored in stack segment. *k is stored on heap.

Ans: c

32. Which of the following is/are true

- a) calloc() allocates the memory and also initializes the allocates memory to zero, while memory allocated using malloc() has random data.
- b) malloc() and memset() can be used to get the same effect as calloc().
- c) calloc() takes two arguments, but malloc takes only 1 argument.
- d) Both malloc() and calloc() return 'void *' pointer.

Ans d

33. Which one of the following is correct syntax for opening a file.

- a) FILE *fopen(const *filename, const char *mode)
- b) FILE *fopen(const *filename)
- c) FILE *open(const *filename, const char *mode)
- d) FILE open(const*filename)

Ans a

34. What is the function of the mode 'w+'?

- a) create text file for writing, discard previous contents if any
- b) create text file for update, discard previous contents if any
- c) create text file for writing, do not discard previous contents if any
- d) create text file for update, do not discard previous contents if any

Ans b

35. _____ removes the named file, so that a subsequent attempt to open it will fail.

- a) remove(const *filename)
- b) remove(filename)

- c) remove()
- d) fclose(filename)

Ans a

36. What is the function of FILE *tmpfile(void)?

- a) creates a temporary file of mode "wb+"
- b) creates a temporary file of mode "wb"
- c) creates a temporary file of mode "w"
- d) creates a temporary file of mode "w+"

Ans a

37. What does tmpfile() returns when it could not create the file?

- a) stream and NULL
- b) only stream
- c) only NULL
- d) does not return anything

Ans a

38. what is the function of fputs()?

- a) read a line from a file
- b) read a character from a file
- c) write a character to a file
- d) write a line to a file

Ans d

39. Which function will return the current file position for stream?

- a) fgetpos()
- b) fseek()
- c) ftell()
- d) fsetpos()

Ans c

40. Which functions is declared in <errno. h>?

- a) fseek()
- b) ftell()
- c) ferror()
- d) fsetpos()

Ans c

41. The _____ function reads atmost one less than the number of characters specified by size from the given stream and it is stored in the string str.

- a) fget()
- b) fgets()
- c) fput()

d) fputs()

Ans b

42. What does the following C code snippet mean?

```
int ungetc(int c, FILE *stream)
```

- a) pushes c back onto a stream
- b) deletes c from the stream
- c) reads frequency of c in stream
- d) no action is taken by the command

Ans A

43. What will be the output of the following C code?

```
1. #include <stdio.h>
2.     union
3.     {
4.         int x;
5.         char y;
6.     }p;
7.     int main()
8.     {
9.         p.y = 60;
10.        printf("%d\n", sizeof(p));
11.    }
```

- a) Compile time error
- b) sizeof(int) + sizeof(char)
- c) Depends on the compiler
- d) sizeof(char)

Ans C

44. How many bytes in memory taken by the following C structure?

```
1. #include <stdio.h>
2.     struct test
3.     {
4.         int k;
5.         char c;
6.     };
```

- a) Multiple of integer size
- b) integer size+character size
- c) Depends on the platform
- d) Multiple of word size

Ans A

45. What will be the output of the following C code?

```
1. #include <stdio.h>
2.     struct student
3.     {
4.         char *c;
5.     };
6.     void main()
7.     {
```

```
8.      struct student s[2];
9.      printf("%d", sizeof(s));
10.     }
```

- a) 2
- b) 4
- c) 16
- d) 8

Ans D

46. Local variables are stored in an area called _____

- a) Heap
- b) Permanent storage area
- c) Free memory
- d) Stack

Ans D

47. Which of the following header files must necessarily be included to use dynamic memory allocation functions?

- a) stdlib.h
- b) stdio.h
- c) memory.h
- d) dos.h

Ans A

48. Which of the following is an incorrect syntax for pointer to structure?

(Assuming struct temp{int b;} *my_struct;)

- a) *my_struct.b = 10;
- b) (*my_struct).b = 10;
- c) my_struct->b = 10;
- d) Both *my_struct.b = 10; and (*my_struct).b = 10;

Ans A

49. Which of the following return-type cannot be used for a function in C?

- a) char *
- b) struct
- c) void
- d) none of the mentioned

Ans d

50. What is the sequence for preprocessor to look for the file within <>?

- a) The predefined location then the current directory
- b) The current directory then the predefined location
- c) The predefined location only
- d) The current directory location

Ans A

51. EOF is an integer type defined in stdio. hand has a value _____

- a) 1
- b) 0
- c) NULL
- d) - 1

Ans d

UNIT - 5 [CLO 5][CLO6]

PART B

4 MARKS

1. What is structure? Write the syntax for structure.
 2. Write the various operations on structure.
 3. How the members of structure object is accessed?
 4. Write the use of size operator on structure
 5. What is a nested structure?
 6. How typedef is used in structure?
 7. Interpret the term Union in C.
 8. Pointout the meaning of Dynamic memory allocation.
 9. Specify the use of typedef.
 10. Generalize the operators used in access the structure members.
 11. Discover the meaning of Array of structure.
 12. Show the difference between Structure from Array.
 13. Invent the application of size of operator to this structure.
- Consider the declaration:
- ```
struct
{
char name;
int num;
} student;
```
14. If we have structure B nested inside structure A, when do we declare structure B?
  15. Discover the meaning of enum. .
  16. Discriminate copying and comparing operation on structures.
  17. Show an example program using union
  18. Mention any two ways passing structures and functions.
  19. What is meant by Self-referential structures
  20. Demonstrate about structures, array of structures and nested structures.
    21. Write a C program using structures to prepare the students mark statement.
    22. Write short note on structure declaration
    23. Referencing pointer to another address to access the memory
    24. Describe about the functions and structures.
    25. Write a C program using structures to prepare the employee pay roll of a company.
    26. Discriminate between malloc and calloc.
    27. Write a C Program to find the Size of a File.
    28. Identify the different types of file.
  29. Compare the terms Field, Record and File.
  30. How to read and write the file.?
  31. Compare fseek() and ftell() function.
  32. What does argv and argc indicate in command-line arguments?
  33. Write the Steps for Processing a File



**UNIT -5 [CLO 5][CLO6]**

**PART C**

**12 MARKS**

1. Write a structure to store the name, account number and balance of customers (more than 10) and store their information. Write a function to print the names of all the customers having balance less than \$200. .
2. Compare and contrast between Structures and Unions. .
3. Examine the differences between static and dynamic memory allocation.
4. Write a C program illustration of subscripted structure variables , arrays of structures.
5. Write a C program for Nested Structure Database Program.
6. Develop a C Program to find the largest number using dynamic memory allocation.
7. Explain in detail about unions with suitable example. .
8. What are Structures? Create a structure with data members of various types and declare two structure variables. Write a program to read data into these and print the same. .
9. Develop a C Program to use the arrays inside union variables. .
10. Write the C coding for finding the average of number stored in sequential access file.
11. Develop a C Program to find the number of lines in a text file. .
12. Write a C Program to calculate the factorial of a number by using the command line argument.
13. Write a C Program to generate Fibonacci series by using command line arguments.