C Programming MCQ

Q1 What is recursion in C programming?

a) It is a function that calls itself

b) It is a function that uses loops

c) It is a function that uses switch case

d) None of the above

Answer: a) It is a function that calls itself

Q2 Which of the following is true about recursion?

a) It is faster than iteration

b) It is slower than iteration

c) It takes more memory than iteration

d) It takes less memory than iteration

Answer: c) It takes more memory than iteration

Q3 1. Which of the following is the base case in recursion?

a) The condition that is checked in every iteration

b) The condition that is checked only once

c) The condition that is not checked

d) None of the above

Answer: b) The condition that is checked only once

Q4 1. Which of the following is a user-defined data type in C?

a) int

b) float

c) double

d) struct

Answer: d) struct

Q5 1. What is the size of a struct with no members in C?

a) 0

b) 1

c) 2

d) 4

Answer: a) 0

Q6 1. Which operator is used to access members of a struct in C?

a) .

b) ->

c) &

d) *

Answer: a) .

Q7 1. What is a union in C?

a) A data structure that allows you to store different data types in the same memory location

b) A collection of related data items

c) A set of rules for how data is stored in memory

d) None of the above

Answer: a

Q8 1. Which of the following is a user-defined data type in C?

a) int

b) float

c) double

d) enum

Answer: d

Q9 1. Which of the following headers do you need to include for file handling in C?

a) stdio.h

b) file.h

c) input.h

d) output.h

Answer: a) stdio.h

Q10 1. What is the default mode when opening a file in C?

a) read mode

b) write mode

c) append mode

d) binary mode

Answer: a) read mode

Q11 What is the output of the following recursive function when called with an argument of 3?

```
void printNum(int n) {
   if (n < 0) {
      return;
```

```
    }
    printf("%d ", n);
    printNum(n-1);
}
```

A. 0 1 2

B. 1 2 3

C. 3 2 1

D. 3 2 1 0

Q12 What is the output of the following code?

```
#include <stdio.h>

struct person {
   char name[20];
   int age;
   float salary;
};

int main() {
   struct person p = { "John", 25, 2500.50 };
   printf("%s %d %.2f\n", p.name, p.age, p.salary);
   return 0;
}
```

A. John 25 2500.50

B. John 25 2500

C. John 2500 25.50

D. Error

Correct answer: A

Q13 What does the typedef keyword do when used with a structure?

a. Defines a new type based on an existing structure type
b. Declares a new structure type with a new name
c. Assigns a value to a structure member
d. Initializes a structure variable
Correct answer: a

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum colors { RED, GREEN, BLUE };
int main() {
    enum colors c = BLUE;
    printf("%d", c);
    return 0;
}
```

a) 0
b) 1
c) 2
d) 3
Answer: c

Q15 What is the output of the following program?

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

union data {
   int i;
   float f;
   char str[20];
};

int main() {
   union data d;

   d.i = 10;
   printf("d.i = %d\n", d.i);

   d.f = 3.14;
   printf("d.f = %f\n", d.f);
```

```
  strcpy(d.str, "Hello World");
  printf("d.str = %s\n", d.str);

  return 0;
}
```

a) d.i = 10, d.f = 3.14, d.str = "Hello World"

b) d.i = 10, d.f = 3.14, d.str = garbage value

c) Compiler error

d) None of the above

Correct answer: a

Q.1. Is it true that too many recursive calls may result into stack overflow?

A) Yes

B) No

Q.2. Which of the following statement is True?

A) User has to explicitly define the numeric value of enumerations

B) User has a control over the size of enumeration variables.

C) Enumeration can have an effect local to the block, if desired

D) Enumerations have a global effect throughout the file.

Q.3. Nested unions are allowed.

A) True

B) False

Q.4. In which case union is better than structure?

A) Less memory is available

B) Faster compilation is required

C) When functions are included

D) None of these

Q.5. Which of the following is the collection of different data types?

A) structure

B) string

C) array

D) All of the above

Q.6. Which keyword is used to define a union?

A) un

B) union

C) Union

D) None of these

Q.7. The '->' operator can be used to access structures elements using a pointer to a structure variable only.

A) True

B) False

Q.8. Which is the correct syntax to declare a file pointer in C?

A) File *file_pointer;

B) FILE *file_pointer;

C) File file_pointer;

D) FILE *file_pointer;

Q.9. Which function is used to seek the file pointer position in C?

A) seek()

B) fseek()

C) fileseek()

D) fmove()

Q.10. Is the following declaration acceptable?

typedef long no, *ptrtono;

no n;

ptrtono p;

A) Yes

B) NO

Q.1. What will be the output of the program?

```
#include<stdio.h>
int sumdig(int);
int main()
{
    int a, b;
    a = sumdig(123);
    b = sumdig(123);
    printf("%d, %d\n", a, b);
    return 0;
}
int sumdig(int n)
{
    int s, d;
    if(n!=0)
    {
        d = n%10;
        n = n/10;
        s = d+sumdig(n);
    }
```

```
    else
        return 0;
    return s;
}
```

A) 4, 4

B) 3, 3

C) 6, 6

D) 12, 12

Q.2. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    struct value
    {
        int bit1:1;
        int bit3:4;
        int bit4:4;
    }bit={1, 2, 13};
    printf("%d, %d, %d\n", bit.bit1, bit.bit3, bit.bit4);
    return 0;
}
```

A) 1, 2, 13

B) 1, 4, 4

C) -1, 2, -3

D) -1, -2, -13

Q.3. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    struct emp
    {
        char *n;
        int age;
    };
    struct emp e1 = {"Dravid", 23};
    struct emp e2 = e1;
    strupr(e2.n);
    printf("%s\n", e1.n);
    return 0;
}
```

A) Error: Invalid structure assignment

B) DRAVID

C) Dravid

D) No output

Q.4. Point out the error in the program?

```
struct emp
{
    int ecode;
    struct emp *e;
};
```

A) Error: in structure declaration

B) Linker Error

C) No Error

D) None of above

Q.5. Point out the error in the program?

```
#include<stdio.h>
int main()
{
    struct emp
    {
        char name[20];
        float sal;
    };
    struct emp e[10];
    int i;
    for(i=0; i<=9; i++)
        scanf("%s %f", e[i].name, &e[i].sal);
    return 0;
}
```

A) Error: invalid structure member

B) Error: Floating point formats not linked

C) No error

D) None of above

Q1 What function is used to open a file in C?

a) read()

b) write()

c) fopen()

d) fclose()

Answer: c) fopen()

Q2 What mode should be used to open a file for writing in C?

a) "r"

b) "w"

c) "a"

d) "b"

Answer: b) "w"

Q3 What is the purpose of the fseek() function in C file handling?

a) to read a specific line in a file

b) to write to a specific location in a file

c) to move the file pointer to a specific location in a file

d) to close a file

Answer: c) to move the file pointer to a specific location in a file

Q4 Which keyword is used to declare a union in C?

a. union

b. struct

c. typedef

d. None of the above

Answer: a

Q5 What is the size of a union in C?

a. The size of the largest data type in the union

b. The sum of the sizes of all the data types in the union

c. The size of the smallest data type in the union

d. None of the above

Answer: a

Q6 What is the purpose of an enumeration in C?

a. To define a set of related constants with names

b. To define a new data type that can hold multiple values

c. To define a new data structure

d. None of the above

Answer: a

Q7 Which keyword is used to define a struct in C?

A) typedef

B) define

C) struct

D) union

Answer: C) struct

Q8 What is the default access specifier for members of a struct in C?

A) public

B) private

C) protected

D) There is no default access specifier

Answer: A) public

Q9 Which of the following is the general format of a recursive function?

a) return_type function_name(arguments){
function_name(arguments);
}

b) return_type function_name(arguments){
if(base_case)
return some_value;
function_name(arguments);
}

c) return_type function_name(arguments){
if(base_case)
return some_value;
return function_name(arguments);
}

d) None of the above

Answer: c) return_type function_name(arguments){
if(base_case)
return some_value;
return function_name(arguments);
}

Q10 What is tail recursion?

a) When the recursive call is the last statement in the function

b) When the recursive call is the first statement in the function

c) When the recursive call is in the middle of the function

d) None of the above

Answer: a) When the recursive call is the last statement in the function

Q11 What is the output of the following recursive function when called with an argument of 2?

```
void printPattern(int n) {
    if (n == 0) {
        return;
    }
    printf("%d ", n);
    printPattern(n-1);
    printf("%d ", n);
}
```

A. 1 2 2 1

B. 2 1 1 2

C. 2 1 2 1

D. 1 2 1 2

Q12 What is the output of the following code?

```
#include <stdio.h>

struct point {
   int x;
   int y;
};

int main() {
   struct point p1 = { 5, 10 };
   struct point p2 = p1;

   printf("(%d,%d) (%d,%d)\n", p1.x, p1.y, p2.x, p2.y);
```

```
    p2.x = 15;

    printf("(%d,%d) (%d,%d)\n", p1.x, p1.y, p2.x, p2.y);

    return 0;
}
```

A. (5,10) (5,10) (5,10) (15,10)

B. (5,10) (5,10) (15,10) (15,10)

C. (5,10) (15,10) (5,10) (15,10)

D. (15,10) (5,10) (15,10) (5,10)

Correct answer: A

Q13 Which of the following is an advantage of using typedef with a structure?

a. It makes the code easier to read and understand

b. It reduces the size of the structure

c. It adds a new member to the structure

d. It makes the structure immutable

Correct answer: a

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum days { SUN, MON, TUE, WED=7, THU, FRI, SAT };
int main() {
    enum days d = TUE;
    printf("%d", THU);
    return 0;
}
```

a) 2

b) 7

c) 8

d) 9

Answer: b

Q15 What is the output of the following program?

```
#include <stdio.h>

union data {
  int i;
  char c;
};

int main() {
  union data d;

  d.i = 65;
  printf("d.i = %d ", d.i);
  printf("d.c = %c ", d.c);

  d.c = 'A';
  printf("d.i = %d ", d.i);
  printf("d.c = %c ", d.c);

  return 0;
}
```

a) d.i = 65 d.c = A d.i = 65 d.c = A

b) d.i = 65, d.c = A, d.i = garbage value, d.c = garbage value

c) Compiler error

d) None of the above

Correct answer: a

Q.1. Value of EOF in C is _____.

A) -1

B) 0

C) 1

D) Null

Q.2. 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

Q.3. Bit fields CANNOT be used in union.

A) True

B) False

Q.4. The '.' operator can be used access structure elements using a structure variable.

A) True

B) False

Q.5. It is not possible to create an array of pointer to structures.

A) True

B) False

Q.6. Which function checks the end-of-file indicator for the given stream in C?

A) eof()

B) EOF

C) feof()

D) None of the above

Q.7. Which of these is a user-defined data type in C?

A) int

B) union

C) char

D) All of these

Q.8. Which function is used to open a file in C?

A) open()

B) fopen()

C) file_open()

D) fileopen()

Q.9. Which of the below statements is incorrect in case of union?

A) Union is a user-defined data structure

B) All data share same memory

C) Union stores methods too

D) union keyword is used to initialize

Q.10. In the following code what is 'P'?

typedef char *charp;

const charp P;

A) P is a constant

B) P is a character constant

C) P is character type

D) None of above

Q.1. Which of the following statements are correct about the program?

```
#include<stdio.h>
int main()
{
    printf("%p\n", main());
    return 0;
}
```

A) It prints garbage values infinitely

B) Runs infinitely without printing anything

C) Error: main() cannot be called inside printf()

D) No Error and print nothing

Q.2. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    struct value
    {
        int bit1:1;
        int bit3:4;
        int bit4:4;
    }bit;
    printf("%d\n", sizeof(bit));
    return 0;
}
```

A) 1

B) 2

C) 4

D) 9

Q.3. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    struct byte
    {
        int one:1;
    };
    struct byte var = {1};
    printf("%d\n", var.one);
    return 0;
}
```

A) 1

B) -1

C) 0

D) Error

Q.4. Point out the error in the program?

```
typedef struct data mystruct;
struct data
{
    int x;
    mystruct *b;
};
```

A) Error: in structure declaration

B) No Error

C) Linker Error

D) None of above

Q.5. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    struct emp
    {
        char *n;
        int age;
    };
    struct emp e1 = {"Dravid", 23};
    struct emp e2 = e1;
    strupr(e2.n);
    printf("%s\n", e1.n);
    return 0;
}
```

A) Error: Invalid structure assignment

B) DRAVID

C) Dravid

D) No output

Q1 Which of the following is true about tail recursion?

a) It is faster than normal recursion

b) It is slower than normal recursion

c) It takes less memory than normal recursion

d) It takes more memory than normal recursion

Answer: c) It takes less memory than normal recursion

Q2 1. Which of the following is the correct way to write a tail recursive function?

a)return_type function_name(arguments){

if(base_case)

return some_value;

function_name(arguments);

}

b) return_type function_name(arguments, int acc){

if(base_case)

return acc;

return function_name(arguments, acc);

}

c) return_type function_name(arguments, int acc){

if(base_case)

return acc;

function_name(arguments, acc);

}

d) None of the above

Answer: b) return_type function_name(arguments, int acc){

if(base_case)

return acc;

return function_name(arguments, acc);

}

Q3 What is the maximum number of recursive calls that can be made in a program?

a) It depends on the size of the stack
b) It depends on the size of the heap
c) It is unlimited
d) None of the above

Answer: a) It depends on the size of the stack

Q4 1. Which of the following is an example of a nested struct in C?

A) struct point { int x; int y; };

B) struct student { char name[20]; int age; };

C) struct rectangle { struct point top_left; struct point bottom_right; };

D) None of the above

Answer: C) struct rectangle { struct point top_left; struct point bottom_right; };

Q5 1. Which of the following is a valid way to initialize a struct in C?

A) struct s = {1, 2, 3};

B) struct s = { .x=1, .y=2 };

C) struct s = {1, .y=2};

D) All of the above

Answer: D) All of the above

Q6 1. What is the keyword used to allocate memory dynamically for a struct in C?

A) malloc

B) free

C) realloc

D) none of the above

Answer: A) malloc

Q7 1. What is the default value of the first element of an enumeration in C?

a. 0

b. 1

c. -1

d. None of the above

Answer: a

Q8 1. 1. Can two elements of an enumeration have the same value in C?

a. Yes

b. No

c. Case specific

d. Enum is not a datatype

Answer: a

Q9 1. What function is used to write a string to a file in C?

a) fprintf()

b) fputc()

c) fputs()

d) fread()

Answer: c) fputs()

Q10 1. What function is used to read a line of text from a file in C?

a) fread()

b) fgetc()

c) fgets()

d) fprintf()

Answer: c) fgets()

Q11

What is the output of the following recursive function when called with an argument of 4?

```
int sum(int n) {
   if (n == 0) {
      return 0;
   }
   return n + sum(n-1);
}
```

A. 10

B. 16

C. 20

D. 24

Q12 What is the output of the following code?

```
#include <stdio.h>

struct person {
  char name[20];
  int age;
  float salary;
};

int main() {
  struct person p1 = { "John", 25, 2500.50 };
  struct person p2 = { "Mary", 30, 3000.00 };

  p1 = p2;
```

```
    printf("%s %d %.2f\n", p1.name, p1.age, p1.salary);


    return 0;
}
```

A. Mary 30 3000.00

B. John 25 2500.50

C. Mary 25 2500.50

D. Error

Correct answer: A

Q13 Which of the following statements declares a typedef for a structure called student?

a. typedef student { int id; char name[50]; }

b. struct student { int id; char name[50]; }

c. typedef struct { int id; char name[50]; } student

d. typedef { int id; char name[50]; } student

Correct answer: c

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum months { JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC };
int main() {
    enum months m1 = APR, m2 = NOV;
    if(m1 == m2) printf("Equal");
    else printf("Not equal");
    return 0;
}
```

a) Equal

b) Not equal

c) Compiler error

d) Runtime error

Answer: b

Q15 What is the output of the following program?

```
#include <stdio.h>

union data {
  int i;
  float f;
  char c;
};

int main() {
  union data d;

  d.i = 65;
  printf("d.i = %d\n", d.i);

  d.f = 3.14;
  printf("d.f = %f\n", d.f);

  d.c = 'A';
  printf("d.c = %c\n", d.c);

  return 0;
}
```

a) d.i = 65, d.f = 3.14, d.c = A

b) d.i = garbage value, d.f = 3.14, d.c = A

c) d.i = 65, d.f = garbage value, d.c = A

d) Compiler error

Correct answer: a

Q.1. In C all functions except main() can be called recursively.

A) True

B) False

Q.2. All members of union ___.

A) Stored in consecutive memory location

B) Share same memory location

C) Store at different location

D) All of these

Q.3. Which of the following statements correct about the below code?

maruti.engine.bolts=25;

A) Structure bolts is nested within structure engine.

B) Structure engine is nested within structure maruti.

C) Structure maruti is nested within structure engine.

D) Structure maruti is nested within structure bolts.

Q.4. size of union is size of the longest element in the union.

A) Yes

B) No

Q.5. Union elements can be of different sizes.

A) True

B) False

Q.6. Are the properties of i, j and x, y in the following program same?

typedef unsigned long int uli;

uli i, j;

unsigned long int x, y;

A) Yes

B) No

Q.7. A recursive function in C ___.

A) Call itself again and again

B) Loop over a parameter

C) Return multiple values

D) None of these

Q.8. one of elements of a structure can be a pointer to the same structure.

A) True

B) False

Q.9. What is the value of EOF in C?

A) -1

B) 0

C) 1

D) Null

Q.10. In the following code, the P2 is Integer Pointer or Integer?(set1)

typedef int *ptr;

ptr p1, p2;

A) Integer

B) Integer pointer

C) Error in declaration

D) None of above

Q.1. What will be the output of the program?

```
#include<stdio.h>
int i;
int fun();
int main()
{
    while(i)
```

```
    {
        fun();
        main();
    }
    printf("Hello\n");
    return 0;
}
int fun()
{
    printf("Hi");
}
```

A) Hello

B) Hi Hello

C) No output

D) Infinite loop

Q.2. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    union a
    {
        int i;
        char ch[2];
    };
    union a u;
    u.ch[0]=3;
    u.ch[1]=2;
```

```
    printf("%d, %d, %d\n", u.ch[0], u.ch[1], u.i);
    return 0;
}
```

A) 3, 2, 515

B) 515, 2, 3

C) 3, 2, 5

D) 515, 515, 4

Q.3. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    enum days {MON=-1, TUE, WED=6, THU, FRI, SAT};
    printf("%d, %d, %d, %d, %d, %d\n", MON, TUE, WED, THU, FRI, SAT);
    return 0;
}
```

A) -1, 0, 1, 2, 3, 4

B) -1, 2, 6, 3, 4, 5

C) -1, 0, 6, 2, 3, 4

D) -1, 0, 6, 7, 8, 9

Q.4. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    enum days {MON=-1, TUE, WED=6, THU, FRI, SAT};
    printf("%d, %d, %d, %d, %d, %d\n", ++MON, TUE, WED, THU, FRI, SAT);
```

```
    return 0;
}
```

A) -1, 0, 1, 2, 3, 4

B) Error

C) 0, 1, 6, 3, 4, 5

D) 0, 0, 6, 7, 8, 9

Q. 5. Point out the error in the program?

```
#include<stdio.h>
int main()
{
    struct a
    {
        float category:5;
        char scheme:4;
    };
    printf("size=%d", sizeof(struct a));
    return 0;
}
```

A) Error: invalid structure member in printf

B) Error in this float category:5; statement

C) No error

D) None of above

Q1 What function is used to write a single character to a file in C?

a) fprintf()

b) fputc()

c) fputs()

d) fread()

Answer: b) fputc()

Q2 What function is used to read a block of data from a file in C?

a) fread()

b) fgetc()

c) fgets()

d) fprintf()

Answer: a) fread()

Q3 What is the return value of the fopen() function if the file cannot be opened?

a) NULL

b) 0

c) -1

d) 1

Answer: a) NULL

Q4 Which of the following is not a valid operation on an enumeration in C?

a. Incrementing

b. Decrementing

c. Multiplying

d. None of the above

Answer: c

Q5 What is the purpose of a typedef in C?

a. To create a new data type with a different name

b. To define a new variable

c. To declare a new function

d. None of the above

Answer: a

Q6 Which keyword is used to declare a typedef in C?

a. typedef

b. struct

c. union

d. None of the above

Answer: a

Q7 Which of the following is an example of a union in C?

A) union { int x; float y; };

B) union { char a; char b; };

C) union { int x; char y; };

D) All of the above

Answer: D) All of the above

Q8 What is the size of a union in C?

A) The size of the largest member

B) The size of the smallest member

C) The sum of the sizes of all members

D) None of the above

Answer: A) The size of the largest member

Q9 What is the purpose of a helper function in recursion?

a) It is used to make the code shorter

b) It is used to make the code faster

c) It is used to avoid using global variables

d) None of the above

Answer: c) It is used to avoid using global variables

Q10 Which of the following is an example of a recursive function?

a) Factorial function

b) Fibonacci function

c) Tower of Hanoi function

d) All of the above

Answer: d) All of the above

Q11 What is the output of the following recursive function when called with an argument of 4?

```
void printBinary(int n) {
        if (n == 0) {
        return;
        }
        printBinary(n/2);
        printf("%d", n%2);
}
```

A. 101

B. 111

C. 100

D. 110

Q12 What is the output of the following code?

```
#include <stdio.h>
struct student {
  char name[20];
  int age;
};

int main() {
  struct student s1 = { "John", 20 };
  struct student s2 = s1;

  s2.age = 25;

  printf("%s %d\n", s1.name, s1.age);
  printf("%s %d\n", s2.name, s2.age);

  return 0;
}
```

A. John 20 John 25

B. John 20 John 20

C. John 25 John 25

D. Error

Correct answer: A

Q13 How can you declare a variable of the student type after using typedef to define the structure?

a. student s;

b. struct student s;

c. int s;

d. char s[50];

Correct answer: a

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum { A, B, C = 1, D, E };
int main() {
    printf("%d", E);
    return 0;
}
```

a) 2

b) 3

c) 4

d) 5

Answer: b

Q15 What is the output of the following program?

```
#include <stdio.h>

union data {
   int i;
   float f;
};

int main() {
   union data d = {10};

   printf("d.i = %d\n", d.i);
   printf("d.f = %f\n", d.f);

   return 0;
}
```

a) d.i = 10, d.f = 0.000000

b) d.i = 10, d.f = garbage value

c) Compiler error

d) None of the above

Correct answer: a

Q.1. typedef's have the advantage that they obey scope rules, that is they can be declared local to a function or a block whereas #define's always have a global effect.

A) Yes

B) No

Q.2. Which function is used to close an opened file in C?

A) close()

B) fclose()

C) file_close()

D) fileclose()

Q.3. A union cannot be nested in a structure.

A) True

B) False

Q.4. The size of a union is ___.

A) Sum of sizes of all members

B) Predefined by the compiler

C) Equal to size of largest data type

D) None of these

Q.5. Which is the correct syntax to create a union?

A) union union_name {

};

B) union union_name {

}

C) union union_name (

);

D) Union union_name (

)

Q.6. The elements of union are always accessed using & operator.

A) Yes

B) No

Q.7. A structure can be nested inside another structure.

A) True

B) False

Q.8. A structure can contain similar or dissimilar elements.

A) True

B) False

Q.9. Usually recursion works slower than loops.

A) Yes

B) No

Q.10. Which function is used to delete an existing file in C?

A) delete()

B) fremove()

C) frem()

D) remove()

Q.1. What will following code do?

```
#include<stdio.h>
int reverse(int);
int main()
{
    int no=5;
    reverse(no);
    return 0;
}
int reverse(int no)
{
    if(no == 0)
        return 0;
    else
        printf("%d,", no);
    reverse (no--);
}
```

A) Print 5, 4, 3, 2, 1

B) Print 1, 2, 3, 4, 5

C) Print 5, 4, 3, 2, 1, 0

D) Infinite loop

Q.2. What will be the output of the program?

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

```
{
    union var
    {
        int a, b;
    };
    union var v;
    v.a=10;
    v.b=20;
    printf("%d\n", v.a);
    return 0;
}
```

A) 10

B) 20

C) 30

D) 0

Q.3. What will be the output of the program?

```
#include<stdio.h>
int main()
{
    enum status {pass, fail, absent};
    enum status stud1, stud2, stud3;
    stud1 = pass;
    stud2 = absent;
    stud3 = fail;
    printf("%d %d %d\n", stud1, stud2, stud3);
    return 0;
}
```

A) 0, 1, 2

B) 1, 2, 3

C) 0, 2, 1

D) 1, 3, 2

Q.4. What will be the output of the program?

```
#include<stdio.h>
    struct course
    {
        int courseno;
        char coursename[25];
    };
int main()
{
    struct course c[] = { {102, "Java"},
                          {103, "PHP"},
                          {104, "DotNet"}     };

    printf("%d ", c[1].courseno);
    printf("%s\n", (*(c+2)).coursename);
    return 0;
}
```

A) 103 DotNet

B) 102 Java

C) 103 PHP

D) 104 DotNet

Q.5. Point out the error in the program?

```
struct emp
```

```
{
    int ecode;
    struct emp e;
};
```

A) Error: in structure declaration

B) Linker Error

C) No Error

D) None of above

Q1 What is the time complexity of a recursive function that calls itself n times?

a) O(n)

b) O(log n)

c) O(n log n)

d) O(n^2)

Answer: a) O(n)

Q2 Which of the following is true about recursion in C programming?

a) It can be used to solve problems that are difficult to solve using iteration

b) It can be used to solve all problems

c) It is always faster than iteration

d) None of the above

Answer: a) It can be used to solve problems that are difficult to solve using iteration

Q3 Which of the following is not required in a recursive function?

a) Base case

b) Recursive call

c) Loop

d) All of the above are required

Answer: c) Loop

Q4 1. Which operator is used to access members of a union in C?

A) .

B) ->

C) &

D) *

Answer: A) .

Q5 1. Which of the following is an example of a typedef declaration for a struct in C?

A) typedef int my_int;

B) typedef struct { int x; int y; } point;

C) typedef float my_float;

D) None of the above

Answer: B) typedef struct { int x; int y; } point;

Q6 1. What is the keyword used to define an enumeration in C?

A) enum

B) define

C) typedef

D) struct

Answer: A) enum

Q7 1. What is the syntax for declaring a typedef for a union in C?

a. typedef union { } union_name;

b. typedef struct { } struct_name;

c. typedef enum { } enum_name;

d. None of the above

Answer: a

Q8 1. 1. What is the purpose of a union in C?

a. To save memory by allowing multiple data types to use the same memory location

b. To group related data items together

c. To define a new data type that can hold multiple values

d. None of the above

Answer: a

Q9 1. What is the return value of the fwrite() function if an error occurs?

a) -1

b) 0

c) 1

d) NULL

Answer: b) 0

Q10 1. What is the purpose of the feof() function in C file handling?

a) to check if the end of the file has been reached

b) to check if an error occurred while reading the file

c) to check if the file is open

d) to check if the file is writable

Answer: a) to check if the end of the file has been reached

Q11 What is the output of the following recursive function when called with an argument of 3?

```
void printSeries(int n) {
        if (n == 0) {
        return;
        }
        printSeries(n-1);
        printf("%d ", n*n);
}
```

A. 1 4 9

B. 9 4 1

C. 1 9 4

D. 4 1 9

Correct answer: A (1 4 9)

Q12 What is the output of the following code?

```
#include <stdio.h>

struct person {
  char name[20];
  int age;
};

void printPerson(struct person *p) {
  printf("%s %d\n", p->name, p->age);
```

```
}

int main() {
   struct person p1 = { "John", 25 };
   struct person *p2 = &p1;

   printPerson(p2);

   return 0;
}
```

A. John 25

B. John

C. 25

D. Error

Correct answer: A

Q13 Which of the following is an example of accessing a member of a structure variable called s using the . operator?

a. s->id

b. s.id

c. *s.id

d. &s.id

Correct answer: b

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum status { FAIL, PASS };
int main() {
    enum status s1 = FAIL, s2 = PASS;
    printf("%d %d", s1, s2);
    return 0;
}
```

a) 0 1

b) 1 0

c) Compiler error

d) Runtime error

Answer: a

Q15 What is the output of the following program?

```
#include <stdio.h>

union data {
  int i;
  float f;
  char c[4];
};

int main() {
  union data d = {.c = "abc"};

  printf("d.i = %d\n", d.i);
  printf("d.f = %f\n", d.f);
  printf("d.c = %s\n", d.c);

  return 0;
}
```

a) d.f = 0.000000, d.c = abc

b) d.i = garbage value, d.f = garbage value, d.c = abc

c) Compiler error

d) None of the above

Correct answer: a

Q1 1. What is the purpose of the rewind() function in C file handling?

a) to close a file

b) to move the file pointer to the beginning of the file

c) to move the file pointer to the end of the file

d) to delete the contents of the file

Answer: b) to move the file pointer to the beginning of the file

Q2 1. What is the purpose of the fflush() function in C file handling?

a) to close a file

b) to clear the buffer for a file stream

c) to write data to a file

d) to read data from a file

Answer: b) to clear the buffer for a file stream

Q3 1. What is the syntax for closing a file in C?

a) close(file);

b) fclose(file);

c) close_file(file);

d) fclose(file);

Answer: b) fclose(file);

Q4 1. What is the syntax for accessing a member of a union in C?

a. union_name.member_name

b. union_name->member_name

c. union_name[member_name]

d. None of the above

Answer: a

Q5 1. What is the purpose of the sizeof operator in C?

a. To determine the size of a data type or variable in bytes

b. To declare a new variable

c. To define a new data type

d. None of the above

Answer: a

Q6 1. Which of the following statements is true about unions in C?

a. Only one member of a union can be used at a time

b. All members of a union can be used at the same time

c. Union members can be of different sizes

d. Union members must be of the same size

Answer: a

Q7 1. Which of the following is an example of an enumeration in C?

A) enum { RED, GREEN, BLUE };

B) enum { 1, 2, 3 };

C) enum { "Monday", "Tuesday", "Wednesday" };

D) None of the above

Answer: A) enum { RED, GREEN, BLUE};

Q8 1. What is the keyword used to define a bit field in a struct in C?

A) bit

B) int

C) char

D) typedef

Answer: B) int

Q9 1. Which of the following is not required in a recursive function?

a) Base case

b) Recursive call

c) Loop

d) All of the above are required

Answer: c) Loop

Q10 1. What is the time complexity of a recursive function that calls itself n times?

a) O(n)

b) O(log n)

c) O(n log n)

d) O(n^2)

Answer: a) O(n)

Q11 What is the output of the following recursive function when called with an argument of 6?

```
int sumDigits(int n) {
        if (n == 0) {
        return 0;
        }
        return n%10 + sumDigits(n/10);
}
```

A. 6

B. 15

C. 21

D. 30

Q12 What is the output of the following code?

```
#include <stdio.h>

struct point {
  int x;
  int y;
};

void printPoint(struct point p) {
  printf("(%d,%d)\n", p.x, p.y);
}

int main() {
  struct point p1 = { 5, 10 };
  struct point p2 = { 15, 20 };

  printPoint(p1);
  printPoint(p2);

  return 0;
}
```

A. (5,10) (15,20)

B. (15,20) (5,10)

C. (5,20) (15,10)

D. Error

Correct answer: A

Q13 What is the correct syntax for declaring a pointer to a student structure called s_ptr?

a. student *s_ptr;
b. struct *s_ptr student;
c. *student s_ptr;
d. struct student **s_ptr;
Correct answer: a

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum numbers { ONE, TWO, THREE };
int main() {
    enum numbers n = TWO;
    switch(n) {
        case ONE: printf("1"); break;
        case TWO: printf("2"); break;
        case THREE: printf("3"); break;
    }
    return 0;
}
```

a) 1
b) 2
c) 3
d) Compiler error
Answer: b

Q15 What is the output of the following program?

```
#include <stdio.h>

struct point {
   int x;
   int y;
};

int main() {
   struct point points[3] = {{1, 2}, {3, 4}, {5, 6}};

   for (int i = 0; i < 3; i++) {
      printf("Point %d: (%d, %d)\n", i+1, points[i].x, points[i].y);
```

```
    }

    return 0;
}
```

a) The program prints the coordinates of 3 points on the screen: (1, 2), (3, 4), and (5, 6).

b) The program produces an error during compilation.

c) The program compiles successfully but does not produce any output.

d) The program produces a runtime error.

Correct answer: a

Q1 Which of the following is true about recursion?

a) It is faster than iteration

b) It is slower than iteration

c) It takes more memory than iteration

d) It takes less memory than iteration

Answer: c) It takes more memory than iteration

Q2 What is tail recursion?

a) When the recursive call is the last statement in the function

b) When the recursive call is the first statement in the function

c) When the recursive call is in the middle of the function

d) None of the above

Answer: a) When the recursive call is the last statement in the function

Q3 What is the maximum number of recursive calls that can be made in a program?

It depends on the size of the stack

It depends on the size of the heap

It is unlimited

None of the above

Answer: a) It depends on the size of the stack

Q4 1. What is the maximum number of bits that can be used in a bit field in C?

A) 8

B) 16

C) 32

D) There is no maximum limit

Answer: D) There is no maximum limit

Q51. Which of the following is an example of a bit field in a struct in C?

A) struct { int x: 4; int y: 4; };

B) struct { char a; char b; };

C) struct { int x; int y; };

D) None of the above

Answer: A) struct { int x: 4; int y: 4; };

Q6 1. Which of the following is a valid way to declare a pointer to a struct in C?

A) struct *p;

B) struct p;

C) struct **p;

D) struct mystruct *p;

Answer: D) struct mystruct *p;

Q7 1. Which of the following statements is true about enumerations in C?

a. Enumerations can be used to define integer constants with meaningful names

b. Enumerations can be used to define floating-point constants with meaningful names

c. Enumerations can be used to define string constants with meaningful names

d. Enumerations can be used to define boolean constants with meaningful names

Answer: a

Q8 1. Which of the following keywords is used to define a constant in an enumeration in C?

a. enum

b. typedef

c. const

d. None of the above

Answer: d

Q9 1. What is the syntax for closing a file in C?

a) close(file);

b) fclose(file);

c) close_file(file);

d) fclose(file);

Answer: b) fclose(file);

Q10 1. What is the purpose of the remove() function in C file handling?

a) to delete a file

b) to rename a file

c) to create a new file

d) to open a file

Answer: a) to delete a file

Q11 What is the output of the following recursive function when called with an argument of 3?

```
int mystery(int n) {
        if (n <= 0) {
        return 0;
        }
        return mystery(n-1) + n;
}
```

A. 1

B. 2

C. 3

D. 6

Q12 What is the output of the following code?

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

int main() {
  struct point p1 = { 5, 10 };
  struct point p2 = { 15, 20 };

  if (p1.x == p2.x && p1.y == p2.y) {
        printf("Points are equal\n");
  }
```

```
  else {
        printf("Points are not equal\n");
  }

  return 0;
}
```

A. Points are equal

B. Points are not equal

C. Error

D. None of the above

Correct answer: B

Q13 What does the sizeof operator return when used with a typedef structure?

a. The size of the structure type

b. The size of a variable of the structure type

c. The size of a pointer to the structure type

d. The size of the first member of the structure type

Correct answer: b

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum { A = -1, B, C = 2 };
int main() {
   printf("%d %d %d", A, B, C);
   return 0;
}
```

a) -1 0 1

b) -1 0 2

c) -1 1 2

d) Compiler error

Answer: b

Q15 What is the output of the following program?

```
#include <stdio.h>

struct employee {
  char name[50];
  int age;
  float salary;
};

int main() {
 struct employee employees[3] = {{"John Doe", 25, 5000.0}, {"Jane Smith", 30, 7000.0}, {"Bob Johnson", 35, 9000.0}};

  FILE *fp = fopen("employees.txt", "w");
  fwrite(employees, sizeof(struct employee), 3, fp);
  fclose(fp);

  fp = fopen("employees.txt", "r");
  struct employee e[3];
  fread(e, sizeof(struct employee), 3, fp);
  fclose(fp);

  for (int i = 0; i < 3; i++) {
    printf("Employee %d: %s %d %.2f\n", i+1, e[i].name, e[i].age, e[i].salary);
  }

  return 0;
}
```

a) The program writes the information of 3 employees to a file named employees.txt and then reads it back and prints the information on the screen.

b) The program produces an error during compilation.

c) The program compiles successfully but does not produce any output.

d) The program produces a runtime error.

Correct answer: a

Q1 1. What is the purpose of the rename() function in C file handling?

a) to delete a file

b) to rename a file

c) to create a new file

d) to open a file

Answer: b) to rename a file

Q2 1. What is the purpose of the ftell() function in C file handling?

a) to return the current position of the file pointer

b) to return the size of the file

c) to return the last error code

d) to return the current date and time

Answer: a) to return the current position of the file pointer

Q3 1. What is the purpose of the fseek() function in C file handling?

a) to move the file pointer to a specific location

b) to close a file

c) to read a line of text from a file

d) to write data to a file

Answer: a) to move the file pointer to a specific location

Q4 1. Which of the following is not a valid operation on a union in C?

a. Assigning a value to one member and reading from another member

b. Assigning a value to one member and reading from the same member

c. Assigning a value to all members at the same time

d. None of the above

Answer: c

Q5 1. What is the syntax for declaring an enumeration in C with named constants?

a. enum {constant1, constant2, constant3};

b. enum {constant1=1, constant2=2, constant3=3};

c. enum {1=constant1, 2=constant2, 3=constant3};

d. enum (constant1, constant2, constant3);

Answer: b

Q6 1. Which of the following headers do you need to include for file handling in C?

a) stdio.h

b) file.h

c) input.h

d) output.h

Answer: a) stdio.h

Q7 1. Which of the following is an example of a function that returns a struct in C?

A) int add(int a, int b);

B) float multiply(float a, float b);

C) struct point create_point(int x, int y);

D) None of the above

Answer: C) struct point create_point(int x, int y);

Q8 1. Which of the following is a user-defined data type in C?

a. int

b. float

c. double

d. enum

Answer: d

Q9 1. 1. What is the purpose of a helper function in recursion?

a) It is used to make the code shorter

b) It is used to make the code faster

c) It is used to avoid using global variables

d) None of the above

Answer: c) It is used to avoid using global variables

Q10 What is the maximum number of recursive calls that can be made in a program?

a) It depends on the size of the stack

b) It depends on the size of the heap

c) It is unlimited

d) None of the above

Answer: a) It depends on the size of the stack

Q11 What is the output of the following recursive function when called with an argument of 3?

```
int mystery(int n) {
        if (n <= 0) {
        return 0;
```

```
        }
        return mystery(n-1) * 2 + 1;
}
```

A. 1

B. 3

C. 7

D. 15

Q12 What is the output of the following code?

```
#include <stdio.h>

struct point {
  int x;
  int y;
};

void printPoint(struct point *p) {
  printf("(%d,%d)\n", p->x, p->y);
}

int main() {
  struct point p1 = { 5, 10 };
  struct point *p2 = &p1;
  p2->x = 15;
  p2->y = 20;

  printPoint(&p1);
```

```
  return 0;
}
```

A. (5,10)

B. (15,20)

C. (5,20)

D. Error

Correct answer: B

Q13 What is the purpose of the . operator when used with a structure variable?

a. To access the address of the structure variable

b. To access a member of the structure variable

c. To allocate memory for the structure variable

d. To free memory for the structure variable

Correct answer: b

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum fruits { APPLE, ORANGE, MANGO };
int main() {
   enum fruits f = APPLE;
   if(f > ORANGE) printf("Yes");
   else printf("No");
   return 0;
}
```

a) Yes

b) No

c) Compiler error

d) Runtime error

Answer: b

Q15 Which of the following is true about passing structures as function arguments in C?

a) Structures cannot be passed as arguments to functions.

b) Structures can only be passed as pointers to functions.

c) Structures can be passed as values or pointers to functions.

d) Structures can only be passed as arrays to functions.

Correct answer: c

Q1 1. Which of the following is true about the stack in recursion?

a) It is used to store the return addresses

b) It is used to store the local variables

c) It is used to store the function arguments

d) All of the above

Answer: d) All of the above

Q2 1. What is the time complexity of a recursive function that calls itself n times?

a) O(n)

b) O(log n)

c) O(n log n)

d) O(n^2)

Answer: a) O(n)

Q3 1. Which operator is used to access members of a struct in C?

A) .

B) ->

C) &

D) *

Answer: A) .

Q4 1. What is the maximum number of members a struct can have in C?

A) 64

B) 128

C) 256

D) There is no maximum limit

Answer: D) There is no maximum limit

Q5 1. What is the keyword used to allocate memory dynamically for a struct in C?

A) malloc

B) calloc

C) realloc

D) free

Answer: A) malloc

Q6 1. What is the keyword used to define an enumeration in C?

A) enum

B) define

C) typedef

D) struct

Answer: A) enum

Q7 1. What is the maximum number of bits that can be used in a bit field in C?

A) 8

B) 16

C) 32

D) There is no maximum limit

Answer: D) There is no maximum limit

Q8 1. Which of the following is a valid way to declare a pointer to a struct in C?

A) struct *p;

B) struct p;

C) struct **p;

D) struct mystruct *p;

Answer: D) struct mystruct *p;

Q9 1. Which keyword is used to declare a union in C?

a. union

b. struct

c. typedef

d. None of the above

Answer: a

Q10 1. Can two elements of an enumeration have the same value in C?

a. Yes

b. No

c. Case specific

d. Enum is not a datatype

Answer: a

Q11 What is the output of the following recursive function when called with an argument of 3?

```
int qux(int n) {
        if (n > 0) {
```

```
        qux(n-1);
        printf("%d ", n);
        qux(n-2);
        }
        return n;
}
```

A. 1 2 3 1

B. 3 2 1 1

C. 1 3 2 3

D. 2 1 3 2

Q12 What is the output of the following code?

```
#include <stdio.h>

struct person {
  char *name;
  int age;
};

int main() {
  struct person p1 = { "John", 25 };
  struct person p2 = p1;
  p2.name = "Jane";

  printf("%s\n", p1.name);

  return 0;
}
```

A. John

B. Jane

C. Error

D. None of the above

Correct answer: A

Q13 Which of the following statements is true about the typedef keyword?

a. It can only be used with structures

b. It creates a new name for an existing type

c. It modifies the structure members

d. It declares a new variable

Correct answer: b

Q14 What will be the output of the following C code?

```
#include <stdio.h>
enum weekdays { MON, TUE, WED, THU, FRI, SAT, SUN };
int main() {
    enum weekdays w1 = TUE, w2 = SUN;
    printf("%d %d", w1, w2);
    return 0;
}
```

a) 1 5

b) 1 6

c) 1 4

d) 1 3

Q15 What is the output of the following program?

```
#include <stdio.h>

union test {
   int x;
   char c;
};

int main() {
   union test t;
```

```
  t.x = 65;
  printf("%c\n", t.c);
  return 0;
}
```

a) A

b) 65

c) The program produces an error during compilation.

d) The program produces a runtime error.

Correct answer: a

Q1 What is the purpose of a typedef in C?

a. To create a new data type with a different name

b. To define a new variable

c. To declare a new function

d. None of the above

Answer: a

Q2 What is the syntax for declaring a typedef for a union in C?

a. typedef union { } union_name;

b. typedef struct { } struct_name;

c. typedef enum { } enum_name;

d. None of the above

Answer: a

Q3 What is the syntax for accessing a member of a union in C?

a. union_name.member_name

b. union_name->member_name

c. union_name[member_name]

d. None of the above

Answer: a

Q4 1. Which of the following statements is true about unions in C?

a. Only one member of a union can be used at a time

b. All members of a union can be used at the same time

c. Union members can be of different sizes

d. Union members must be of the same size

Answer: a

Q5 1. What is the default mode when opening a file in C?

a) read mode

b) write mode

c) append mode

d) binary mode

Answer: a) read mode

Q6 1. What function is used to open a file in C?

a) read()

b) write()

c) fopen()

d) fclose()

Answer: c) fopen()

Q7 1. What function is used to write a string to a file in C?

a) fprintf()

b) fputc()

c) fputs()

d) fread()

Answer: c) fputs()

Q8 1. Which of the following is true about recursion?

a) It is faster than iteration

b) It is slower than iteration

c) It takes more memory than iteration

d) It takes less memory than iteration

Answer: c) It takes more memory than iteration

Q9 1. Which of the following is true about tail recursion?

a) It is faster than normal recursion

b) It is slower than normal recursion

c) It takes less memory than normal recursion

d) It takes more memory than normal recursion

Answer: c) It takes less memory than normal recursion

Q10 What is the maximum number of recursive calls that can be made in a program?

It depends on the size of the stack

It depends on the size of the heap

It is unlimited

None of the above

Answer: a) It depends on the size of the stack

Q11 What is the output of the following recursive function when called with an argument of 5?

```
int baz(int n) {
        if (n == 0) {
        return 1;
        }
        return 2 * baz(n-1);
}
```

A. 1

B. 2

C. 16

D. 32

Q12 What is the output of the following code?

```
#include <stdio.h>

struct person {
  char *name;
  int age;
};

int main() {
  struct person p1 = { "John", 25 };
  struct person p2 = p1;
  p2.age = 30;

  printf("%d\n", p1.age);
```

```
  return 0;
}
```

A. 25

B. 30

C. Error

D. None of the above

Correct answer: A

Q13 What is the difference between a struct and a typedef struct declaration?

a. There is no difference

b. A typedef struct declaration creates a new type name for the structure

c. A struct declaration creates a new type name for the structure

d. A typedef struct declaration allows access to private members of the structure

Correct answer: b

Q14 What is the output of the following code snippet?

```
#include <stdio.h>
enum { A = 10, B, C = 5, D };
int main() {
   printf("%d %d %d %d", A, B, C, D);
   return 0;
}
```

a) 10 11 5 6

b) 10 11 5 5

c) 10 10 5 6

d) 10 10 5 5

Answer: a

Q15 What is the output of the following program?.

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

struct book {
```

```
  char title[50];
  int pages;
  float price;
};

int main() {
  struct book my_book = {"A Tale of Two Cities", 307, 12.50};
  printf("%d\n", strlen(my_book.title));
  return 0;
}
```

a) 21

b) 22

c) 20

d) None of the above

Correct answer: c