**C Programming – MCQ Practice Set**

**Data Types and Basics**

1. What is the size of float on most 32-bit systems?  
   A. 2 bytes B. 4 bytes C. 6 bytes D. 8 bytes
2. What is the result of sizeof('A') in C?  
   A. 1 B. 2 C. 4 D. Compiler dependent
3. Which of the following is not a valid data type in C?  
   A. int B. float C. bool D. char
4. What is the default value of a local variable in C?  
   A. 0 B. Garbage value C. NULL D. Compiler dependent
5. What does the keyword void indicate?  
   A. No value returned B. Null pointer C. Empty memory D. No variable

**Operators and Expressions**

1. What is the result of 7 % -4?  
   A. 3 B. -3 C. -1 D. 1
2. Which operator has the highest precedence?  
   A. + B. \* C. = D. ()
3. What will a = b = c = 10; result in?  
   A. a = 10 only B. b = 10 only C. All variables = 10 D. Error
4. What is the output of printf("%d", 10 + 2 \* 3);?  
   A. 36 B. 26 C. 16 D. 30
5. Which operator is used to allocate memory dynamically?  
   A. & B. \* C. malloc D. sizeof

**Control Structures**

1. How many times will the following loop execute?

for(int i = 0; i < 5; i++);

A. 0 B. 4 C. 5 D. Infinite

1. What is the output?

int x = 5;

if (x = 0)

printf("Zero");

else

printf("Non-zero");

A. Zero B. Non-zero C. Error D. None

1. Which of these is not a valid loop in C?  
   A. for B. while C. do-while D. loop
2. What is break used for?  
   A. Skip loop B. Exit loop immediately C. Continue next iteration D. End program
3. What is the correct syntax for a switch case?  
   A. switch case B. switch(value) C. switch = value D. case switch

**Functions**

1. What is the default return type of a function in C?  
   A. void B. int C. float D. char
2. Which keyword is used to pass arguments by reference?  
   A. ref B. & C. \* D. Not directly supported
3. What is the output?

int fun() {

static int count = 0;

return ++count;

}

If fun() is called 3 times, what is the result of the last call?  
A. 1 B. 2 C. 3 D. 4

1. What will this print?

void test() {

return 5;

}

A. 5 B. Error C. Garbage D. void

1. What is a function prototype?  
   A. Memory allocation B. Function definition C. Declaration before use D. Return statement

**Pointers**

1. What does the following declaration mean?

int \*ptr;

A. ptr is an int B. ptr is a pointer to an int C. int is a pointer D. ptr is a reference

1. What is a NULL pointer?  
   A. Pointer to zero B. Pointer to invalid memory C. Pointer to nowhere D. All of the above
2. What is the output?

int a = 5;

int \*p = &a;

printf("%d", \*p);

A. 5 B. Address C. Error D. 0

1. Which operator is used to access value at a pointer address?  
   A. & B. \* C. -> D. %
2. What is ptr++ when ptr is a pointer to int?  
   A. Increment address by 1 byte B. 4 bytes C. 2 bytes D. Error

**Strings**

1. What is the output?

char str[] = "Hello";

printf("%d", sizeof(str));

A. 5 B. 6 C. 7 D. Error

1. Which function is used to copy strings?  
   A. strcpy() B. memcpy() C. strcat() D. strmove()
2. Which function reads a line of input?  
   A. gets() B. scanf() C. fgets() D. getch()
3. What is the output of strlen("C Language")?  
   A. 10 B. 9 C. 11 D. 8
4. What terminates a string in C?  
   A. '\n' B. '\r' C. NULL D. '\0'

**Structures & Unions**

1. Which keyword defines a structure?  
   A. union B. struct C. class D. record
2. Can structures contain arrays?  
   A. No B. Yes C. Only static arrays D. Only pointers
3. How do you access members of a structure using a pointer?  
   A. . B. -> C. & D. \*
4. What is the output?

struct s {

int x;

}s1 = {5};

printf("%d", s1.x);

A. 5 B. x C. Error D. Undefined

1. What is true about unions?  
   A. Can store multiple values B. Memory shared by all members C. Each member gets own memory D. They use dynamic allocation

**Arrays**

1. What is the index of the first element in a C array?  
   A. 1 B. 0 C. -1 D. Depends
2. Can you change the size of an array once declared?  
   A. Yes B. No C. Only if global D. Only in dynamic memory
3. What is int a[5] = {1};?  
   A. Initializes all to 1 B. First to 1, rest to 0 C. Error D. Garbage values
4. What is the output?

int arr[] = {10, 20, 30};

printf("%d", \*(arr + 1));

A. 10 B. 20 C. 30 D. Error

1. What is the result of accessing arr[1000] in a 5-element array?  
   A. 0 B. Segmentation fault C. 1000 D. Compiler error

**Memory Management**

1. Which function frees dynamically allocated memory?  
   A. malloc() B. calloc() C. free() D. release()
2. Which function sets allocated memory to zero?  
   A. malloc() B. calloc() C. realloc() D. memset()
3. What is the output?

int \*p = malloc(sizeof(int) \* 5);

A. Allocates 5 bytes B. Allocates space for 5 ints C. Allocates 1 int D. Compiler error

1. What is memory leak?  
   A. Using extra memory B. Allocated memory not freed C. Invalid pointer D. Using freed pointer
2. Can you return local array from function?  
   A. Yes B. No C. Only with global arrays D. Only with static arrays

**Miscellaneous & Advanced**

1. Which header file is required for printf()?  
   A. conio.h B. stdlib.h C. stdio.h D. string.h
2. Can a function return a pointer?  
   A. Yes B. No C. Only with global pointer D. Only for arrays
3. What is typedef used for?  
   A. Create function B. Rename data type C. Initialize struct D. Define union
4. What is the maximum depth of nested loops allowed in C?  
   A. 15 B. 127 C. No limit D. Compiler dependent
5. What does static keyword do inside a function?  
   A. Makes variable global B. Keeps value between function calls C. Speeds execution D. Makes memory constant

**Answer Key – C Programming MCQs**

**Data Types and Basics**

1. B
2. C
3. C
4. B
5. A

**Operators and Expressions**

1. C
2. D
3. C
4. C
5. C

**Control Structures**

1. C
2. B
3. D
4. B
5. B

**Functions**

1. B
2. D
3. C
4. B
5. C

**Pointers**

1. B
2. D
3. A
4. B
5. B

**Strings**

1. B
2. A
3. C
4. A
5. D

**Structures & Unions**

1. B
2. B
3. B
4. A
5. B

**Arrays**

1. B
2. B
3. B
4. B
5. B

**Memory Management**

1. C
2. B
3. B
4. B
5. D

**Miscellaneous & Advanced**

1. C
2. A
3. B
4. D
5. B