

C LANG TEST-6 (FUNCTIONS)

Total points 50/50 ?

STUDENT NAME *

VIVA

✓ 1. A function in C is a block of code that *

1/1

- ☐ A) Executes once only
- ☐ B) Performs a specific task
- ☐ C) Executes only if called
- ☒ D) Both B and C



✓ 2. The keyword used to define a function in C is *

1/1

- ☐ A) define
- ☐ B) function
- ☒ C) void or data type
- ☐ D) fun



✓ 3. Every C program must have at least one function — *

1/1

- ☒ A) main()
- ☐ B) printf()
- ☐ C) scanf()
- ☐ D) return()



✓ 4. Function declaration is also called *

1/1

- ☒ A) Function prototype
- ☐ B) Function body
- ☐ C) Function call
- ☐ D) Function definition



✓ 5. A function that calls itself is called *

1/1

- ☐ A) Nested function
- ☒ B) Recursive function
- ☐ C) Inline function
- ☐ D) Self function



✓ 6. The part of the program where a variable can be used is called its * 1/1

- ☒ A) Scope
- ☐ B) Range
- ☐ C) Lifetime
- ☐ D) Access area



✓ 7. A function that does not return any value must be declared as * 1/1

- ☐ A) int
- ☒ B) void
- ☐ C) float
- ☐ D) null



✓ 8. Functions help to * 1/1

- ☐ A) Reduce code length
- ☐ B) Reuse code
- ☐ C) Make code modular
- ☒ D) All of the above



✓ 9. Which of the following correctly defines a function? *

1/1

- ☒ A) void fun(int a, int b);
- ☐ B) fun(int a, int b);
- ☐ C) define fun(int a, int b);
- ☐ D) function fun(a, b);



✓ 10. Function definition contains *

1/1

- ☐ A) Function name only
- ☒ B) Body of the function
- ☐ C) Only declaration
- ☐ D) None



✓ 11. A function call transfers control from *

1/1

- ☐ A) One block to another
- ☒ B) Caller to called function
- ☐ C) Called function to main()
- ☐ D) Header to footer



✓ 12. The value passed to a function is known as *

1/1

- ☐ A) Parameter
- ☒ B) Argument
- ☐ C) Variable
- ☐ D) Constant



✓ 13. Formal parameters are used in *

1/1

- ☐ A) Function call
- ☐ B) Function definition
- ☐ C) Function declaration
- ☒ D) Both B and C



✓ 14. Actual parameters are used in *

1/1

- ☐ A) Function definition
- ☒ B) Function call
- ☐ C) Function prototype
- ☐ D) Function body



✓ 15. The return type of main() is usually *

1/1

- ☐ A) void
- ☒ B) int
- ☐ C) float
- ☐ D) char



✓ 16. The function which does not return a value must have return type *

1/1

- ☒ A) void
- ☐ B) int
- ☐ C) double
- ☐ D) char



✓ 17. A function may return *

1/1

- ☒ A) Only one value
- ☐ B) Multiple values
- ☐ C) Array of values
- ☐ D) None



✓ 18. The return statement is used to *

1/1

- ☐ A) Stop execution
- ☐ B) Exit from a function
- ☐ C) Return control to main()
- ☒ D) Both B and C



✓ 19. Which of these must match between function declaration and definition?

*1/1

- ☐ A) Function name
- ☐ B) Number and type of parameters
- ☐ C) Return type
- ☒ D) All of the above



✓ 20. Which of these is a valid function declaration? *

1/1

- ☐ A) int sum();
- ☐ B) int sum(int, int);
- ☐ C) void sum(void);
- ☒ D) All of these



✓ 21. Functions can be declared *

1/1

- ☐ A) Inside another function
- ☒ B) Outside all functions
- ☐ C) Both A and B
- ☐ D) None



✓ 22. If no value is returned by a function, it should have return type *

1/1

- ☐ A) int
- ☐ B) float
- ☒ C) void
- ☐ D) none



✓ 23. A function call within another function is known as *

1/1

- ☐ A) Recursive call
- ☒ B) Nested call
- ☐ C) Function chaining
- ☐ D) Function overriding



✓ 24. In C, functions can be defined *

1/1

- ☐ A) Before main()
- ☐ B) After main()
- ☒ C) Both A and B
- ☐ D) None



✓ 25. The keyword used to return a value from a function is *

1/1

- ☐ A) exit
- ☐ B) stop
- ☒ C) return
- ☐ D) break



✓ 26. If a function is defined but never called, *

1/1

- ☐ A) It executes automatically
- ☒ B) It is ignored
- ☐ C) It causes an error
- ☐ D) It runs after main()



✓ 27. What happens if a function returns no value but is declared as int? * 1/1

- ☐ A) No error
- ☒ B) Warning
- ☐ C) Runtime error
- ☐ D) Compilation error



✓ 28. Which is not a valid function name in C? * 1/1

- ☐ A) sum
- ☐ B) add_2
- ☒ C) 2add
- ☐ D) total



✓ 29. What is the default return type of a function in old C versions (K&R C)? *1/1

- ☒ A) int
- ☐ B) float
- ☐ C) void
- ☐ D) double



✓ 30. Functions that call themselves directly or indirectly are *

1/1

- ☒ A) Recursive
- ☐ B) Indirect
- ☐ C) Inline
- ☐ D) Dynamic



✓ 31. Recursion must have *

1/1

- ☒ A) Base condition
- ☐ B) Loop
- ☐ C) Pointer
- ☐ D) Array



✓ 32. What is the output of this function? *

1/1

```
int fun() {  
    return 10;  
}
```

- ☐ A) 0
- ☒ B) 10
- ☐ C) Garbage
- ☐ D) Error



✓ 33. Function parameters can be passed by *

1/1

- ☐ A) Value
- ☐ B) Reference
- ☒ C) Both A and B
- ☐ D) None



✓ 34. In call by value, changes made inside the function *

1/1

- ☐ A) Affect original value
- ☒ B) Don't affect original value
- ☐ C) Delete the variable
- ☐ D) None



✓ 35. In call by reference, the address of variable is passed *

1/1

- ☒ A) True
- ☐ B) False



✓ 36. The keyword used to define inline functions is *

1/1

- ☒ A) inline
- ☐ B) static
- ☐ C) fast
- ☐ D) auto



✓ 37. Functions can return arrays directly *

1/1

- ☐ A) True
- ☒ B) False



✓ 38. The main purpose of user-defined functions is *

1/1

- ☒ A) Code reusability
- ☐ B) Slow execution
- ☐ C) More complexity
- ☐ D) Error creation



✓ 39. A recursive function without base condition results in *

1/1

- ☒ A) Infinite recursion
- ☐ B) Compilation error
- ☐ C) Fast execution
- ☐ D) Termination



✓ 40. Functions are compiled *

1/1

- ☐ A) When they are called
- ☒ B) When defined
- ☐ C) At runtime
- ☐ D) Not compiled



✓ 41. The parameters listed in a function declaration are *

1/1

- ☐ A) Actual parameters
- ☒ B) Formal parameters
- ☐ C) Both
- ☐ D) None



✓ 42. The parameters listed in a function call are *

1/1

- ☐ A) Formal parameters
- ☒ B) Actual parameters
- ☐ C) Dummy parameters
- ☐ D) None



✓ 43. The maximum number of arguments a function can take is *

1/1

- ☐ A) 10
- ☐ B) 32
- ☒ C) Depends on compiler
- ☐ D) Unlimited



✓ 44. The function name followed by parentheses is called *

1/1

- ☐ A) Function declaration
- ☒ B) Function call
- ☐ C) Function prototype
- ☐ D) Function header



✓ 45. The function header includes *

1/1

- ☐ A) Return type
- ☐ B) Function name
- ☐ C) Parameter list
- ☒ D) All of the above



✓ 46. What is the output of this function call? *

1/1

```
int add(int a, int b){ return a+b; }  
main(){ printf("%d", add(2,3)); }
```

- ☐ A) 2
- ☐ B) 3
- ☒ C) 5
- ☐ D) Error



✓ 47. A function can be declared multiple times but defined only once. * 1/1

- ☒ A) True
- ☐ B) False



✓ 48. Which function returns control to the operating system? * 1/1

- ☐ A) stop()
- ☐ B) main()
- ☐ C) return 0;
- ☒ D) exit(0);



✓ 49. Function overloading is * 1/1

- ☒ A) Not supported in C
- ☐ B) Supported in C
- ☐ C) Supported with same parameters
- ☐ D) None



✓ 50. Recursive functions use * 1/1

- ☒ A) Stack
- ☐ B) Queue
- ☐ C) Heap
- ☐ D) Array



Google Forms

