|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be output if you compile and execute the following c code?**  void main(){    float a=5.2;    if(a==5.2)      printf("Equal");   else if(a<5.2)      printf("Less than");   else      printf("Greater than");  }   |  | | --- | | Equal | | Less than | | Greater than | | Compiler error | | |
| **2** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be output if you compile and execute the following c code?**  void main(){   int i=4,x;   x=++i + ++i + ++i;   printf("%d",x);  }   |  | | --- | | 21 | | 18 | | 12 | | Compiler Error | | |
| **3** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be output if you compile and execute the following c code?**  void main(){  int a=2;  if(a==2){    a=~a+2<<1;    printf("%d",a);  }  else{   break;  }  }   |  | | --- | | It will print nothing. | | -3 | | -2 | | Compiler error | | |
| **4** | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **What is the error in the following declaration?**  struct outer{ int a; struct inner{ char c; }; };   |  | | --- | | Nesting of structure is not allowed in c. | | It is necessary to initialize the member variable. | | Inner structure must have name. | | Outer structure must have name. | | There is no error | | |
| **5** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be the output if you compile and execute the following C code?**  void main(){   int a=10;   printf("%d %d %d",a,a++,++a);  }   |  | | --- | | 12 11 11 | | 12 10 10 | | 11 11 12 | | 10 10 12 | | |
| **6** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are common character constants?**  1. 'A' 2. 'b' 3. '30' 4. '$'   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | | |
| **7** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_operator is used to combine two conditional expressions that evaluate to true as a whole only if either of the expressions evaluates to true.**   |  | | --- | | && | | || | | & | | ^ | | |
| **8** | |  |  |  | | --- | --- | --- | | **Relational operator always evaluates to the value, either true or false.**   |  | | --- | | True | | False | | |
| **9** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Ternary operator acts as a substitute to the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_statement.**   |  | | --- | | if-else | | switch-case | | if | | if-else and switch-case | | |
| **10** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is not an example of jump statement?**   |  | | --- | | break | | continue | | for | | All of the above | | |
| **11** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is not an example of sorting algorithm in 'C' programming?**   |  | | --- | | Selection Sort | | Bubble Sort | | Insertion Sort | | None of the above | | |
| **12** | |  |  |  | | --- | --- | --- | | **A pointer is a variable which contains stored value of another variable which has been assigned.**   |  | | --- | | True | | False | | |
| **13** | |  |  |  | | --- | --- | --- | | **A function call is the statement which actually invokes the function and passes values to it.**   |  | | --- | | True | | False | | |
| **14** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is not an element of function?**   |  | | --- | | Declaration | | Definition | | Function call | | None of the above | | |
| **15** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **By default local variables are defined in \_\_\_\_\_\_\_\_\_\_\_.**   |  | | --- | | auto | | static | | int | | global | | |
| **16** | |  |  |  | | --- | --- | --- | | **Variables are named location in memory.**   |  | | --- | | True | | False | | |
| **17** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **toupper() function belongs to which of the following Header file?**   |  | | --- | | math.h | | Ctype.h | | Stdlib.h | | stdio.h | | |
| **18** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are the keywords that constitute the storage classes?**  1. auto 2. static 3. extern 4. register   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | | |
| **19** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **In nested loops, which of the following loops takes control of the number of complete repetitions of the inner?**   |  | | --- | | Inner loop | | Outer loop | | main function | | Depends on the program | | |
| **20** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are the features of pointers?**  1. A pointer cannot have the same name as an existing variable. 2. A pointer can contain the address of any of the basic data types in C, as well as arrays and other advanced data structures. 3. A pointer should point only variable of the same data type. 4. A pointer should be assigned an address before it is used in a statement.   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | | |
| **21** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are the special operators used with pointers?**  1. '\*' 2. '&' 3. '.'   |  | | --- | | 1, 2 | | 2, 3 | | 1, 3 | | 1, 2, 3 | | |
| **22** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be result when below code is compiled and executed?**  #include<iostream.h> void main() {       Printf("Hello World"); }   |  | | --- | | Compile and Run successfully without any output on the screen. | | Compile and Run successfully with text “Hello World” on the screen. | | Compile with error. | | Compile successfully but runtime error. | | |
| **23** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Select the correct value of i.**  #include<stdio.h> void main() {      int i = 10;      float f = 7777.50f;      i = f; }   |  | | --- | | 10 | | 7777 | | 7777.500000 | | Error: Type casting is required. | | |
| **24** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **In which of the following situations, pointer can be used?**  1. To return more than one value from a function 2. To pass arrays more conveniently from one function to another 3. To manipulate arrays easily by moving pointers to them, instead of moving the arrays themselves 4. To allocate memory and access it   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | | |
| **25** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **In which of the following situations, pointers can be used?**  1. To return more than one value from a function. 2. To pass arrays more conveniently from one function to another. 3. To manipulate arrays easily by moving pointers to them, instead of moving the arrays themselves. 4. To allocate memory and access it.   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | | |
| **26** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are valid function declarations?**  1. void divide (int a, int b) {return a/b ;} 2. void divide (int a, int b) {int c; c=a/b; return c ;}  3. void divide (int a, int b) {int c; c=a/b; return ;} 4. int divide (a, b) {int a, b, c; c=a/b; return c ;} 5. int divide (int a, int b){return a/b;}   |  | | --- | | 1, 3 | | 2, 5 | | 3, 4 | | 3, 5 | | |
| **27** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following will be the output of the above code?**  1. #include<stdio.h>  2. #define VAR 100 3. void main ( ) { 4. #if (VAR > 75) 5. #if (VAR > 100) 6. printf ("var is greater than 100"); 7. #else 8. printf ("var is greater than 75");  9. #endif 10. }   |  | | --- | | The code will compile successfully and the output will be: var is greater than 100. | | The code will compile successfully and the output will be: var is greater than 75. | | The code will generate compile time error as preprocessor directory is not defined correctly. | | The code will generate a compile time error as the nested if-else construct is not properly implemented. | | |
| **28** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following statements are true?**  1. A preprocessor symbol is never replaced if it occurs within single or double quotations. 2. All preprocessors begin with the number of sharp (#) sign. 3. More than one directives can appear on one line. 4. A comment can be included at the end of a macro definition. 5. Macros can have an abbreviation for two line of text.   |  | | --- | | 1, 2, 3 | | 1, 2, 4 | | 2, 4, 5 | | None of the above | | |
| **29** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be the output of the below code when it is compiled/run?**  1. #include<stdio.h>  2. #define var1 100 #define var2 200 3. void main ( ) { 4. printf ("value of var1 is %d", var1); 5. printf("value of var2 is %d", var2); 6. }   |  | | --- | | The code will compile successfully but generate runtime error as the Macros are not defined properly. | | The code will generate compile time error as preprocessor directory is not defined correctly. | | The code will generate a compile time error at line 5 as the variable is not declared. | | value of 100 is 100 value of 200 is 200 | | |
| **30** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be the output of the above code when it is compiled/run?**  1. #include <stdio.h> 2. #define NAME "SMITH" 3. void main ( ) { 4. #if !defined (MYNAME)  5. printf ("MYNAME IS MARTIN"); 6. #else 7. printf ("MYNAME IS %s", NAME); 8. #endif 9. }   |  | | --- | | The code will generate compile time error as preprocessor directory is not defined correctly. | | The code will compile successfully and output of code will be: MYNAME IS SMITH | | The code will compile successfully but will not display anything. | | The code will generate compile time error at line 4 as the function defined used is invalid. | | |
| **31** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following statement/s is/are correct about Array?**  1. C does not allow an array index of positive integer.  2. The address of an array can be passed as an argument to a function.  3. A function can receive only one dimensional array. If function needs a two dimensional array then such array can be passed in the form of multiple single dimensional arrays.   |  | | --- | | 1 | | 1, 2 | | 1, 3 | | 2 | | |
| **32** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **An array Matrix is defined as int Matrix [3][4];**  Which of the following represents the first and the last element in this Matrix?   |  | | --- | | Matrix[0][0] , Matrix[2][3] | | Matrix[0][0] , Matrix[3][2] | | Matrix[0][1] , Matrix[2][3] | | Matrix[1][1] , Matrix[3][3] | | |
| **33** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following declarations can be used to declare a pointer named "pt" pointing to a variable 'var' of type int?**  1. int \*pt, var; 2. int var, \*pt 3. int pt, var; 4. int\* pt, \* var; 5. int\* pt, var;   |  | | --- | | 1 | | 1, 2, 5 | | 2, 4, 5 | | 3, 5 | | |
| **34** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the below listed assignment/s is/are not valid for the declared variables in first line? (Assume that variable is already initialized).**  float home, first, second;  1. home = & second;  2. home = second;  3. first++; home --;  4. ++\* home; 5. home ++\*2;   |  | | --- | | 1 | | 1, 2 | | 1, 4 | | 3, 5 | | |
| **35** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Read the following statements and choose the correct answer:**  1. During run time the number of nodes in a linked list may vary.  2. Like an array, linked list also has a prefixed storage space in which the elements can be placed.  3. The header and trailer are special pointers of system defined data type that point to the starting and ending node of every linear linked list.   |  | | --- | | 1 is correct | | 2, 3 are correct | | 1, 2 are correct | | 1, 3 are correct | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be the output if you compile and execute the following C code?  void main(){ int i=5,j=2; if(++i>j++||i++>j++) printf("%d",i+j);  }**   |  | | --- | | 7 | | 11 | | 8 | | 9 | | |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **What will be the output if you compile and execute the following C code?**  **#define max 5; void main(){ int i=0; i=max++; printf("%d",i++);  }**   |  | | --- | | 5 | | 6 | | 7 | | 0 | | Compiler error | |   **2** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be the output if you compile and execute the following C code?**  void main(){ char \*str; scanf("%[^\n]",str); printf("%s",str);  }   |  | | --- | | It will accept a word as a string from user. | | It will accept a sentence as a string from the user and display when pressed enter. | | It will accept a paragraph as a string from user. | | Compiler error | |   **3** |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **What will be the output if you compile and execute the following C code?**  **void main(){ int array[3]={5}; int i; for(i=0;i<=2;i++) printf("%d ",array[i]);  }**   |  | | --- | | 5 garbage garbage | | 5 0 0 | | 5 null null | | Compiler error | | None of above | |   **4** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What will be output if you will compile and execute the following c code?**  **void main(){    int i=320;    char \*ptr=(char \*)&i;    printf("%d",\*ptr);  }**   |  | | --- | | 320 | | 1 | | 64 | | Compile Error | |   **5** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following functions is used when strings including blank spaces are to be accepted?**   |  | | --- | | **scanf(%s)** | | **gets()** | | **accept()** | | **All of the above** | |   **6** |
| |  |  |  |  | | --- | --- | --- | --- | | **By default local variables are defined in \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.**   |  | | --- | | **auto** | | **static** | | **extern** | |   **7** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following data types cannot store values with decimal points?**   |  | | --- | | **Integer** | | **Float** | | **Double** | | **None of the above** | |   **8** |
| |  |  |  |  | | --- | --- | --- | --- | | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is unformatted console I/O function.**   |  | | --- | | ***printf()*** | | ***scanf()*** | | ***getchar()*** | |   **9** |
| |  |  |  | | --- | --- | --- | | **Default statement is not compulsory in the switch-case.**   |  | | --- | | **True** | | **False** | |   **10** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following loops guarantees to run at-least once?**   |  | | --- | | **do....while** | | **for......** | | **while......** | | **All of the above** | |   **11** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is correct about the below program?**  **void main()  {     int a=10,b=20;     char x=1,y=0;     if(a,b,x,y)     {        printf("EXAM");     }  }**   |  | | --- | | 'XAM' is printed | | 'EXAM' is printed | | Compile time Error | | Nothing is printed | |   **12** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Array passed as an argument to a function is interpreted as which of the following?**   |  | | --- | | **Address of the array** | | **Values of the first elements of the array** | | **Address of the first element of the array** | | **Number of elements of the array** | |   **13** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is the correct way of declaring a float pointer?**   |  | | --- | | **float ptr;** | | **float \*ptr;** | | **\*float ptr;** | | **None of the above** | |   **14** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_is the file that contains the declaration of functions and pre-processor statements, which help to access the externally defined function.**   |  | | --- | | **Source code** | | **Header file** | | **Object file** | | **Binary executables** | |   **15** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following operations can be performed on pointers?**  **1. Addition 2. Subtraction 3. Multiplication**   |  | | --- | | 1, 2 | | 2, 3 | | 1, 3 | | 1, 2, 3 | |   **16** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is mean Pushdown list?**  **1. Stack 2. Queue 3. Linked list**   |  | | --- | | 1, 2 | | 2, 3 | | 1, 3 | | 1, 2, 3 | |   **17** |
| |  |  |  |  | | --- | --- | --- | --- | | **Character constants should be enclosed between which of the following?**   |  | | --- | | **Single quotes** | | **Double quotes** | | **Both the above** | |   **18** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is the size of the string variable?**   |  | | --- | | **1 byte** | | **4 byte** | | **8 byte** | | **None of the above** | |   **19** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **In which of the following sorting techniques, the values of the elements are compared with the value in the adjacent element and if it is smaller, value is swapped?**   |  | | --- | | **Selection Sort** | | **Bubble Sort** | | **Insertion Sort** | | **All of the above** | |   **20** |
| |  |  |  | | --- | --- | --- | | **Variables are named location in memory.**   |  | | --- | | **True** | | **False** | |   **21** |
| **22** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are the keywords that constitute the storage classes?**  1**. auto 2. static 3. extern 4. register**   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | |   **23** |
| |  | | --- | | **Inner loop** | | **Outer loop** | | **main function** | | **Depends on the program** |  |  | | --- | | **In nested loops, which of the following loops takes control of the number of complete repetitions of the inner?** |   **24** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are the features of pointers?**  1. A pointer cannot have the same name as an existing variable. 2. A pointer can contain the address of any of the basic data types in C, as well as arrays and other advanced data structures. 3. A pointer should point only variable of the same data type. 4. A pointer should be assigned an address before it is used in a statement.   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | |   **25** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following are common character constants?**  1. 'A' 2. 'b' 3. '30' 4. '$'   |  | | --- | | 1, 2, 3 | | 2, 3, 4 | | 1, 3, 4 | | 1, 2, 3, 4 | |   **26** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_operator is used to combine two conditional expressions that evaluate to true as a whole only if either of the expressions evaluates to true.**   |  | | --- | | && | | || | | & | | ^ | |   **27** |
| |  |  |  | | --- | --- | --- | | **Relational operator always evaluates to the value, either true or false.**   |  | | --- | | True | | False | |   **28** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Ternary operator acts as a substitute to the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_statement.**   |  | | --- | | if-else | | switch-case | | if | | if-else and switch-case | |   **29** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is not an example of jump statement?**   |  | | --- | | break | | continue | | for | | All of the above | |   **30** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is not an example of sorting algorithm in 'C' programming?**   |  | | --- | | Selection Sort | | Bubble Sort | | Insertion Sort | | None of the above | |   **31** |
| |  |  |  | | --- | --- | --- | | **A pointer is a variable which contains stored value of another variable which has been assigned.**   |  | | --- | | True | | False | |   **32** |
| |  |  |  |  | | --- | --- | --- | --- | | **“Little Champs” is an English medium school. The teacher started by defining a prime number as a natural number that has exactly two distinct natural number divisors 1 and the number itself. The smallest twenty-five prime numbers are 2,3,5,7, 13,17,19,23, 29,31, 37,41,43,47,53,59,61,67,71,73,79,83,89,97.101**  Which of the following codes will generate the prime numbers between 1 and 100?  A:main() {          int n=100,i=1,j,c;          clrscr();          while (i<=n) {             c=0;             for (j=1;j<=i;j++)   {               if(i%j==0)                 c++;             }            if(c==2)              printf("%d    ",i);                 i++;     } } B:main() {          int n=100,i=1,j,c;          clrscr();          while (i<=n) {             c=0;             for (j=1;j<=i;j++)                  if(i%j==0)                 c++;                         if(c==2)              printf("%d    ",i);                 i++;     } }    C:main() {          int n=100,i=1,j,c;          clrscr();          while (i<=n) {             c=0;             for (j=1;j<=i;j++)   {               if(i%j==1)                 c++;             }            if(c==2)              printf("%d    ",i);                 i++;     } } return 0; }   |  | | --- | | A | | B | | C | |   **33** |
| |  |  |  |  | | --- | --- | --- | --- | | **Common data types include integers, booleans, characters, strings. Given as input an integer number of seconds, print as output the equivalent time in hours, minutes, and seconds. Recommended output format is something like: 7322 seconds is equivalent to 2 hours 2 minutes 2 seconds.**  Which of the following code will do the following:  A. main() { int sec; printf("Please enter number of seconds.""\n"); scanf ("%d", &sec); printf ("%d seconds is equal to ",sec); printf ("%d hours ",sec/3600); printf ("%d minutes ",sec/60); printf ("%d seconds.",sec%60); return 0; }   B main() { int sec; printf("Please enter number of seconds.""\n"); scanf ("%d", &sec) printf ("%d seconds is equal to ",sec); printf ("%d hours ",sec/3600); printf ("%d minutes ",sec/60); printf ("%d seconds.",sec%60) return 0; }  C.main() { int sec; printf("Please enter number of seconds.""\n"); scanf ("%d", &sec); printf ("%d seconds is equal to ",sec); printf ("%d hours ",sec/3600); printf ("%d minutes ",sec/60); printf ("%d seconds.",sec%60);  }   |  | | --- | | A | | B | | C | |   **34** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following statements are FALSE about "C"?**  1. C is not a strongly typed language. 2. C is a block-structured language. 3. C program is divided into units called codes. 4. In C, function name is followed by parentheses and these parentheses must contain one or more parameters.   |  | | --- | | 1, 2 | | 1, 2, 3 | | 2, 3 | | 2, 3, 4 | |   **35** |

|  |  |
| --- | --- |
|  |  |
| **1** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_operator is used to combine two conditional expressions that evaluate to true as a whole only if either of the expressions evaluates to true.**   |  | | --- | | && | | || | | & | | ^ | | |
| **2** | |  |  |  | | --- | --- | --- | | **Relational operator always evaluates to the value, either true or false.**   |  | | --- | | True | | False | | |
| **3** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Ternary operator acts as a substitute to the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_statement.**   |  | | --- | | if-else | | switch-case | | if | | if-else and switch-case | | |
| **4** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is not an example of jump statement?**   |  | | --- | | break | | continue | | for | | All of the above | | |
| **5** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following is not an example of sorting algorithm in 'C' programming?**   |  | | --- | | Selection Sort | | Bubble Sort | | Insertion Sort | | None of the above | | |
| **6** | |  |  |  | | --- | --- | --- | | **A pointer is a variable which contains stored value of another variable which has been assigned.**   |  | | --- | | True | | False | | |
| **7** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Consider the declaration below. Which of the following statements define the variable of type employee? (Turbo C Compiler used.)**  struct employee { int ID;  char name [20];  int salary; };   |  | | --- | | employee e1,e2; | | employee struct e1,e2; | | struct e1,e2; | | struct employee e1,e2; | | |
| **8** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Read the following statements and choose the correct answer:**  1. Just by observing the algorithm of recursive function, it is possible to identify     the local variables which are to be stacked. 2. A recursive algorithm can be converted into a non-recursive algorithm.  3. When the recursive routine returns a value, a returned value and address is saved, the data area is freed, and a branch to the return address is executed. 4. The performance of an operation in several steps, with each step using the output of the preceding step is termed as recursion.   |  | | --- | | 2, 3, 4 are correct | | 1, 2 are correct | | 1, 2, 3, 4 are correct | | 1, 3 are correct | | |
| **9** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **PC memory offset addresses are in the range \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.**   |  | | --- | | from 1 K to 256 K | | from 1 K to 64 K | | from 1 M to 256 M | | from 64K to 1M | | |
| **10** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **When the below code is executed, what will be the output ?**  int a=10,b; b=a++ + ++a; printf("%d,%d,%d,%d", b, a++, a, ++a);   |  | | --- | | 12,10,11,13 | | 12,11,11,11 | | 22,10,11,13 | | 22,13,13,13 | | |
| **11** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Which of the following statements are appropriate with reference to pointers used in C language?**  1. A variable can reference another function's local variable using pointers.  2. Pointers are useful as the content of variables may change during execution of program but its address will not get changed.  3. Formal variables of functions can be accessed using the pointer in other functions.  4. Use of pointers in function reduces the execution time.  5. Use of pointers in function saves the memory space.   |  | | --- | | 1, 2, 3 | | 1, 2, 4 | | 2, 3, 4 | | 3, 4, 5 | | |
| **12** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **What value does testarray[2][1][0] in the sample code below contain?**  int testarray [3][2][2] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12};   |  | | --- | | 11 | | 3 | | 5 | | 7 | | |
| **13** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Referring to the sample below, what is MAX\_NUM?**  # define MAX\_NUM 15   |  | | --- | | MAX\_NUM is a linker constant. | | MAX\_NUM is a precompiler constant. | | MAX\_NUM is a preprocessor macro. | | MAX\_NUM is an integer constant. | | |
| **14** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **The function array\_dup(), defined below, contains an error. Which one of the following statements correctly analyzes it?**  void \* array\_dup (a, number, size)  const void \* a;  size\_t number;  size\_t size;  {  void \* clone;  size\_t bytes;  assert(a != NULL);  bytes = number \* size;  clone = alloca(bytes);  if (!clone)  return clone;  memcpy(clone, a, bytes);  return clone;  }   |  | | --- | | array\_dup () declares its first parameter to be a pointer, when the actual argument will be an array. | | If the arguments to memcpy () refer to overlapping regions, the destination buffer will be subject to memory corruption. | | size\_t is not a Standard C defined type, and may not be known to the compiler. | | The memory obtained from alloca () is not valid in the context of the caller. Moreover, alloca () is nonstandard. | | |
| **15** | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **In the sample code below, what is the value of myArray[1][2]; ?**  int i,j;  int ctr = 0;  int myArray[2][3];  for (i=0; i<  3; i++)  for (j=0; j<  2; j++)  {  myArray[j][i] = ctr;  ++ctr;  }   |  | | --- | | 1 | | 2 | | 3 | | 5 | | |