**Ques.1 Do each of the following: (1\*8 = 8 Marks)**

1. Write the function header for a function called **exchange** that takes two pointers to floating-point

numbers **x** and **y** as parameters and does not return a value.

**b)** Write the function prototype for the function in part (a).

**c)** Write the function header for a function called **evaluate** that returns an integer and that takes as

parameters integer **x** and a pointer to function **poly**. Function **poly** takes an integer parameter and returns

an integer.

**d)** Write the function prototype for the function in part (c).

**e)** Write the function header for function **zero** which takes a long integer array parameter **bigIntegers** and

does not return a value.

**f)** Write the function prototype for the function in part (e).

**g)** Write the function header for function **add1AndSum** which takes an integer array parameter

**oneTooSmall** and returns an integer.

**h)** Write the function prototype for the function described in part (g).

**Ques.2** How many bytes in memory would be occupied by the following array of pointers to strings? How many bytes would be required to store the same strings, if they are stored in a two-dimensional character array? **(2 Marks)**

**char \*mess[ ] = { “Hammer and Tongs”, “Tooth and Nail”, “Spit and Polish”, “ You and C” };**

**Ques.3** Can an array of pointers to strings be used to collect strings from the keyboard? If not, why? **(2 Marks)**

**Ques.4** Show two different methods of initializing character array **vowel** with the string of vowels, **“AEIOU”**. **(2 Marks)**

**Ques.5 State whether the following statements are True/False : (1\*10 = 10 Marks)**

1. When the **printf** function is called it always begins printing at the beginning of a new line.
2. A string when read using scanf() needs an ampersand character.
3. Comments are instructions that cause some action to take place.
4. If S1 and S2 are two strings, then concatenation operation produces a string which contains characters of

S2 followed by the characters of S1.

1. A function cannot be defined inside another function.
2. If return type for a function is not specified, it defaults to int.
3. C considers the variables **number** and **NuMbEr** to be identical.
4. A string **Hello World** can be read using scanf.
5. To ensure that a character is uppercase, the **toupper** conversion function is used.
6. Are the three declarations char \*\*apple, char \*apple[ ], and char apple[ ][ ] same?

**Ques.6** Write a C program to find the length of a given string. (Do not use “strlen” function) **(2 Marks)**

**Ques.7** Write a C program to copy the contents of a given string into another string. (Do not use “strcpy” function)

**(2 Marks)**

**Ques.8 Find the output for the following program segments: (1 \* 8 = 8 Marks)**

1. #include<stdio.h>

void main()

{

int const a = 5;

a++;

printf(“%d”, a);

}

1. #include<stdio.h>

void main()

{

const int a = 5;

a++;

printf(“%d”, a);

}

1. #include<stdio.h>

void main()

{

char s[ ]="hello", t[ ]="hello";

if(s = = t)

printf("equal strings");

else

printf("No Output");

}

1. #include<stdio.h>

void main()

{

int x = ~1;

printf(“%d”, x); }

1. #include<stdio.h>

void f()

{

static int i;

++i;

printf("%d", i);

}

void main()

{

f();

f();

f();

}

1. #include<stdio.h>

void main()

{

register int x = 5;

int \*p;

p=&x;

x++;

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

}

1. Compiler Error
2. 5
3. 6
4. Garbage Value
5. #include<stdio.h>

void f()

{

printf("Hello\n");

}

void main()

{

;

}

1. No output
2. Error, as the function is not called
3. Error, as the function is defined without its declaration
4. Error, as the main() function is left empty
5. #include<stdio.h>

void main()

{

int i = 13, j = 60;

i ^= j;

j ^= i;

i ^= j;

printf("%d %d", i, j);

}

**Ques.9 Fill in the following blanks : (1\*12 = 12 Marks)**

1. The bits in the result of an expression using the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ operator are set to 1 if the corresponding bits in each operand are set to 1. Otherwise, the bits are set to zero.
2. The \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ operator returns the location in memory where its operand is stored.
3. To simulate call by reference when passing a non-array variable to a function, it is necessary to pass the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ of the variable to the function.
4. The conversion specifiers \_\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_\_ are used to print strings and characters, respectively.
5. The bits in the result of an expression using the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ operator are set to 1 if atleast one of the corresponding bits in either operand are set to 1. Otherwise, the bits are set to zero.
6. The process of placing the elements of an array in order is called \_\_\_\_\_\_\_ the array.
7. The \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ operator returns the value of the object to which its operand points.
8. **The elements of an array are related by the fact that they have the same \_\_\_\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_\_.**
9. Variables declared in a block or in the parameter list of a function are assumed to be of storage class \_\_\_\_\_\_\_\_\_\_\_\_\_ unless specified otherwise.
10. The \_\_\_\_\_\_\_\_\_\_\_\_ keyword/statement, when executed in a repetition structure causes the next iteration of the loop to be performed immediately.

**Ques.10 Answer the following with reference to the following program segment: (1\*10 = 10 Marks)**

int i, j = 25;

int \*pi, \*pj = &j;

\*pj = j + 5;

j = \*pj + 5;

pj = pj;

\*pi = i + j;

**Each integer quantity occupies 2 bytes of memory. The value assigned to i begin at (hexadecimal) address F9C and the value assigned to j begins at address F9E. What will be the values of the following:**

1. &i
2. &j
3. pj
4. \*pj
5. **i**
6. pi
7. \*pi
8. (pi + 2)
9. (\*pi + 2)
10. \* (pi + 2)

**Ques.11 Match the following with reference to the following segment: (1\*10 = 10 Marks)**

**int x[3][5] = { {1, 2, 3, 4, 5}, {6, 7, 8, 9, 10}, {11, 12, 13, 14, 15} }, \*n = &x;**

1. \* ( \* (x + 2) + 1 ) a. 9
2. \* ( \* (x + 2) + 4 ) b. 13
3. \* ( \* (x + 1)) c. 4
4. \* ( \* ( x ) + 2 ) + 1) d. 3
5. \* ( \* ( x + 1 ) + 3 ) e. 2
6. \*n f. 12
7. \* ( n + 2 ) g. 14
8. ( \* ( n + 3 ) + 1 ) h. 7
9. \* ( n + 5 ) + 1 i. 1
10. ++ \*n j. 8

k. 5

l. 10

m. 6

**Ques.12 Match the following with reference to the following segment: (1\*10 = 10 Marks)**

**unsigned int arr[3][3] = { 2, 4, 6, 9, 1, 10, 16, 64, 5 };**

1. \*\*arr a. 64
2. \*\*arr < \* ( \*arr + 2 ) b. 18
3. \* ( arr + 2 ) / ( \* ( \*arr + 1 ) > \*\*arr) c. 6
4. \* ( arr[1] + 1 ) | arr[1][2] d. 3
5. \* ( arr[0] ) | \* (arr[2] ) e. 0
6. arr[1][1] < arr[0][1] f. 16
7. arr[2][1] & arr[2][0] g. 1
8. arr[2][2] | arr[0][1] h. 11
9. arr[0][1] ^ arr[0][2] i. 20
10. ++ \*\*arr + --arr[1][1] j. 2

k. 5

l. 4

**Ques.13 What does this program do? (2 Marks)**

#include<stdio.h>

int mystery(const char \*, const char \*);

void main()

{

char string1[80], string2[80];

printf("Enter two strings\n");

scanf("%s%s", string1, string2);

printf("The result is %d\n", mystery(string1,string2));

}

int mystery(const char \*s1, const char \*s2)

{

for( ; \*s1!='\0' && \*s2!='\0'; s1++, s2++)

if(\*s1 != \*s2)

return 0;

return 1;

}

**Ques.14** What is a double pointer? Explain by giving an example. **(2 Marks)**

**Ques.15** What is the full form of ASCII? **(2 Marks)**

**Ques.16** What is the built-in function to append one string to another? **(2 Marks)**

**Ques.17** What is the default value of local and global variables? **(2 Marks)**

**Ques.18** Convert the following into its given equivalent codes. Show steps of conversion also. **(4 Marks)**

1. (84)10 = ( ? )4  **(ii)** (2C9)16 = ( ? )10  **(iii)** (101010)2 =(?)8  **(iv)** (3674)8 = ( ? )2

**Ques.19** Write a C program to sort “n” numbers in an array in ascending order using Selection sort algorithm.

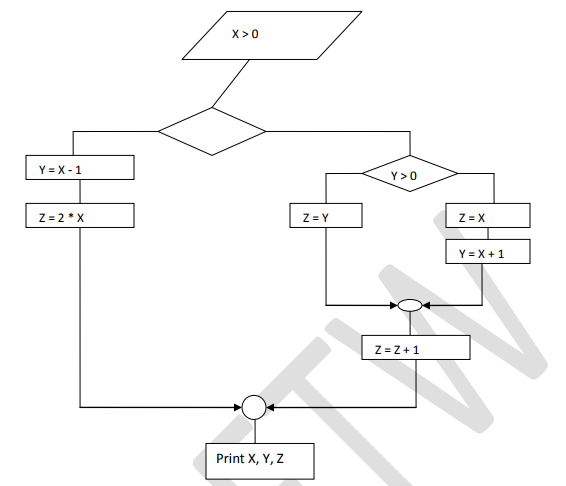
**(3 Marks)**

**Ques.20** Write a C program to sort “n” numbers in an array in ascending order using Bubble sort algorithm.

**(3 Marks)**

**Ques.21** Write code for the flowchart given below. Assume that the variables x and y are integers and z is a float.

**(2 Marks)**



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*