SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

COIMBATORE – 35

DEPARTMENT OF COMPUTER SIENCE AND ENGINEERING (UG & PG)

First Year & 1st Semester

2 Marks Question and Answer

Subject Code & Name: 19CST101 – Programming for Problem Solving

UNIT - III

1. What is an array?

An array is a group of similar data types stored under a common name.

int a[10];

Here a[10] is an array with 10 values.

2. What are the main elements of an array declaration?

Array name
Type and
Size

3. What is dynamic memory allocation?

Allocating the memory at run time is called as dynamic memory allocation.

4. What are the various dynamic memory allocation functions? malloc() - Used to allocate blocks of memory in required size of bytes. free() - Used to release previously allocated memory space. calloc() - Used to allocate memory space for an array of elements. realloac() - Used to modify the size of the previously allocated memory space.

5. Why is it necessary to give the size of an array in an array declaration?

When an array is declared, the compiler allocates a base address and reserves enough space in the memory for all the elements of the array. The size is required to allocate the required space. Thus, the size must be mentioned.

6. Where in memory are variables stored?

Variables can be stored in several places in memory, depending on their lifetime.

(1) Variables that are defined outside any function (whether of global or file static scope), and variables that are defined inside a function as static variables, exist for the lifetime of the program's execution. These variables are stored in the data segment. The data segment is a fixed-size area in memory set aside for these variables.

(2) Variables that are the arguments functions exist only during the execution of that function. These variables are stored on the stack. The stack is an area of memory that starts out as small and grows automatically up to some predefined limit.

(3) The third area is the one that does not actually store variables but can be used to store data pointed to by variables. Pointer variables that are assigned to the result of a call to the function malloc() contain the address of a dynamically allocated area of memory. This memory is in an area called the heap.

7. What is an heap memory?

The heap is another area that starts out as small and grows, but it grows only when the programmer explicitly calls malloc() or other memory allocation functions, such as calloc(). The heap can share a memory segment with either the data segment or the stack, or it can have its own segment, it all depends on the compiler options and operating system. The heap, like the stack, has a limit on how much it can grow, and the same rules apply as to how that limit is determined.

8. What is the difference between an array and pointer?

Array:

1.Array allocates space automatically.
2.It cannot be resized.
3.It cannot be reassigned.
4.Size of(array name) gives the number of bytes occupied by the array.

Pointer:

1.Pointer is explicitly assigned to point to an allocated space.
2.It can be resized using realloc ().
3.Pointers can be reassigned.
4.Sezeof(pointer name) returns the number of bytes used to store the pointer variable

9. Comparison of arrays and structures is as follows.

ARRAY:

An array is a collection of data items of same data type.
Arrays can only be declared.
There is no keyword for arrays.
An array name represents the address of the starting element.
An array cannot have bit fields.

STRUTURE:

A structure is a collection of data items of different data types.
Structures can be declared and defined.
The deyword for structures is struct. A structrure name is known as tag. It is a shorthand notation of the declaration.
A structure may contain bit fields.

10. List the characteristics of Arrays.

All elements of an array share the same name, and they are distinguished form one another with help of an element number. Any particular element of an array can be modified separately without disturbing other elements.

11. What are the types of Arrays?

1.One-Dimensional Array
2. Two-Dimensional Array
3. Multi-Dimensional Array

12. What is the use of ‘\0’ character?

When declaring character arrays (strings), ‘\0’ (NULL) character is automatically added at end. The ‘\0’ character acts as an end of character array.

13. Define Strings.

The group of characters, digit and symnbols enclosed within quotes is called as Stirng (or) character Arrays. Strings are always terminated with ‘\0’ (NULL) character. The compiler automatically adds ‘\0’ at the end of the strings.

14. Write an example for declaring ARRAYS.

Arrays may consist of any of the valid data types. Arrays are declared along with all other variables in the declaration section of the program.

```
/* Introducing array's */
#include <stdio.h>
main() { int
numbers[100]; float
averages[20];
numbers[2] = 10; --
numbers[2];
printf("The 3rd element of array numbers is %d\n", numbers[2]); }
```

15. Write an example for assigning values to ARRAYS.

The declaration is preceded by the word static. The initial values are enclosed in braces,

Example

```
#include <stdio.h>
main() { int x;
static int values[] = { 1,2,3,4,5,6,7,8,9 }; static
char word[] = { 'H','e','l','l','o' };
for( x = 0; x < 9; ++x )
printf("Values [%d] is %d\n", x, values[x]);
}
```

16. Define Multi-dimentional arrary.

Multi-dimensioned arrays have two or more index values which specify the element in the array.

```
multi[i][j];
DECLARATION:
int m1[10][10];
static int m2[2][2] = { {0,1}, {2,3} };
```

17. Example for CHARACTER ARRAYS [STRINGS]

```
#include <stdio.h>
main() {
static char name1[] = {'H','e','l','l','o'};
static char name2[] = "Hello";
printf("%s\n", name1); printf("%s\n",
name2);
}
```