Fundamentals of Computer science

Unit 1

1) A computer program is a program that is a sequence of instructions in a programming language that a computer can execute or interpret. They are defined through the use of syntactic and semantic rules, to determine structure and meaning respectively. Over the years, programming languages have evolved drastically. First is machine level which uses binary notations, then assembly language. Then come the introduction of high level languages like Basic, followed by system programming language like C, and also object oriented languages like C++.

2)

	Compilation	Interpretation
a)	Reads the program completely and gives the result.	Reads the program line by line and gives the result.
b)	Shows each possible error	Shows the error first encountered.

Unit 2

1) ~~Operator~~

In the following table, highest precedence appear at the top and the lowest at the bottom. Higher precedence operator will be evaluated first.

Operator	Associativity
1) [] -> . ++ --	Left to right
+ - ! ~ ++ -- (type) sizeof	Right to left
* / %	Left to right
+ -	Left to right

Operators	Associativity
() [] -> .	Left to right
++ --	Left to right
* / %	Left to right
+ -	Left to right
<< >>	Left to right
< <= > >=	Left to right
== !=	Left to right
&	Left to right
^	Left to right
|	Left to right
&&	Left to right
||	Left to right
?:	~~Left to right~~ Right to left
= += -= *= /= %= >>= <<= &= ^= |=	Right to left
,	Right to left.

2) Break and Continue statement are used to alter control flow. Break terminates the loop and causes execution to resume after the loop. Continue statement causes the next iteration of the loop to run immediately.

Break:

```
while (1)
{
If (n<0) break;
brk (n);
n= n-1;
}
```

Current loop is ~~ended~~ ended and execution picks up with next statement after the loop.

Continue:

```
real num;
num=0;
```

for & n

Continue:

```
#include <stdio.h>
int main() {
  int i;
  double number, sum = 0.0;

  for (i = 1; i <= 10; ++i) {
    printf("Enter a n%.d: ", i);
    scanf("%.1lf", &number);

    if (number < 0.0) {
      Code Continue;
    }
    sum += number;
  }
  printf("Sum = %.2lf", sum);

  return 0;
}
```

Unit 3

1) A preprocessor is one of the automatically used by the compiler to execute certain steps or transform the program before actual compilation. Preprocessor command begins with (#) symbol
Example: #define, #include

2) C language provides a set of pre-defined functions called string handling functions to work with string values. They are defined in a header file called string.h. Commonly used string handling functions are strlen(), strcpy().

```
#include <stdio.h>
int main()
{
int length;
char s[20] = "String";
length = strlen(s);
printf("\length of the string is =%d\n", length);
return 0;
}
```

Unit 4

1) Command line arguments are simple arguments that are specified after the name of the program in the system's command line and these arguments values are passed on to the program during its execution. They are passed to main() function and they control program from outside.

2) Pointer arithmetic: A pointer in C can also be treated like a numeric value, so therefore you can perform add arithmetic operations just like numeric values. The four of arithmetic operators are, ++, --, +, -.

Unit 5

1) Array of structures is an array in which each element is a structure of the same type. Referencing and subscripting of these arrays follow the same rules as does simple array.
Example:-

Eto Storing information of 5 students.

```
#include <stdio.h>
#include <string.h>

struct student {
int rollno;
char name[10];
};

int main() {
 int i;
struct student st[5];
printf("Enter Records of 5 students");
for (i=0; i<5; i++) {
printf ("\n\t Enter Rollno: ");
scanf ("%d", &st[i].rollno);
printf ("\n Enter Name: ");
 scanf("%s", &st[i].name);
}
printf("\n Student List:");
for(i=0; i<5; i++) {
printf("\n Rollno: %d, Name: %s", st[i].rollno, st[i].name);
}
return 0;
}
```

2) typedef is a keyword used in C language to provide names to already existing variables in the program. It redefines the name, in other words.