LAB MANUAL 02

TOKENS & ESCAPE SEQUENCES

Lab Objectives:

At the end of this lab students will know about

- **Tokens**
- **Types of escape sequences**
- **Uses of escape sequences**
- **How to use in C++ programs**
- **What is a data type?**
- **Different types of data types used in C++**
- **How to use each data type**

Tokens

Tokens are the minimal chunk of program that have meaning to the compiler –the smallest meaningful symbols in the language. Our code displays all 6 kinds of tokens, though the usual use of operators is not present here:

Token type	Description/Purpose	Examples
Keywords	Words with special meaning to the compiler	int, double,for, if
Identifiers	Names of sources in which input or any data is stored	cout, x1, cin
Literals	Basic constant values	24.3, “Hello world”
Operators	To perform operations like logical, arithematic etc	&, ||, +, -
Punctuation/Separators	Defining the structure of program	{,},(, ;
Whitespace	Spaces of various types use for formatting	Newline, tab, backslash

Escape sequences:

– a symbol used to represent a special character in a text literal. The \n indicates a newline character. It is an example of an escape sequence. Here are all the C++ escape sequences which you can include in strings:

Escape Sequence	Represented Character
\a	System bell (beep sound)
\b	Backspace
\f	Form feed (page break)
\n	Newline (line break)
\r	“Carriage return” (returns cursor to start of line)
\t	Tab
\\	Backslash
\′	Single quote character
\"	Double quote character

Data Types

Every expression has a type – a formal description of what kind of data its value is. For instance, 0 is an integer, 3.142 is a floating-point (decimal) number, and "Hello, world!\n" is a string value(a sequence of characters). Data of different types take a different amounts of memory to store. Here are the built-in data types we will use most often:

Type Names	**Description**	**Size**	**Range**
char	Single text character or small integer. Indicated with single quotes (’a’, ’3’).	1 byte	signed: -128 to 127 unsigned: 0 to 255
int	Larger integer.	4 byte	signed: -2147483648 to 2147483647 unsigned: 0 to 4294967295
boolean	Boolean (true/false). Indicated with the keywords true and false.	1 byte	Just true (1)or false (0).
double	“Doubly” precise floating point number.	8 byte	+/-1.7e +/-308 (15 digits)

For Example:

```
#include<iostream>
#include<conio.h>
using namespace std;
main()
{
 int x=3;
y=3*x;
cout<< “value of x =”<< x<< “value of y=” << y;
getch();
}
```

Comment [1]: int y=3*x;

Question#01

Print Diamond using escape sequence “\n” and “\t”? In single cout statement?

Output should be like

```
                *

          *     *     *

    *     *     *     *     *

          *     *     *

                *
```

Question #02

Print your Result Card using required escape sequences? Print all other information as it is but get input from user in obtained marks and calculate the percentage of entered marks?

Output:

```
                        Result Card

Name: ABC                                  Reg. #: CIIT/---------

Session: Fall-16                           Semester: 2

Subjects                      Total   Marks           Obtained Marks
```

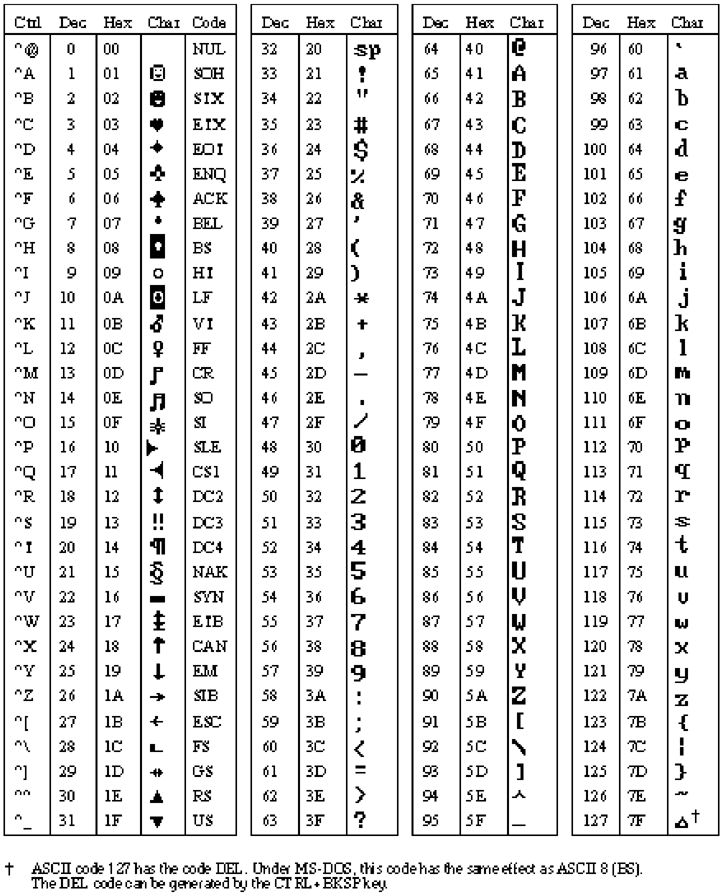
Digital Logic Design	**100**	**___user input___**
Electronics-I	**100**	**___user input___**
English	**100**	**___user input___**

Percentage= %

Question#03
Develop a C++ program that declare almost 6 types of identifiers and display their sizes in bytes on the screen.

Question #04
Develop a C++ program that prints the table of 2 without using loop?

Question #5:
write a program to swap the value of a and b
let a=10 and b = 20
your output should be
a = 20 and b = 10



Ctrl	Dec	Hex	Char	Code	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
^@	0	00		NUL	32	20	sp	64	40	@	96	60	`
^A	1	01	☺	SOH	33	21	!	65	41	A	97	61	a
^B	2	02	☻	STX	34	22	"	66	42	B	98	62	b
^C	3	03	♥	ETX	35	23	#	67	43	C	99	63	c
^D	4	04	♦	EOT	36	24	$	68	44	D	100	64	d
^E	5	05	♣	ENQ	37	25	%	69	45	E	101	65	e
^F	6	06	♠	ACK	38	26	&	70	46	F	102	66	f
^G	7	07	•	BEL	39	27	'	71	47	G	103	67	g
^H	8	08	◘	BS	40	28	(	72	48	H	104	68	h
^I	9	09	○	HT	41	29	)	73	49	I	105	69	i
^J	10	0A	◙	LF	42	2A	*	74	4A	J	106	6A	j
^K	11	0B	♂	VT	43	2B	+	75	4B	K	107	6B	k
^L	12	0C	♀	FF	44	2C	,	76	4C	L	108	6C	l
^M	13	0D	♪	CR	45	2D	-	77	4D	M	109	6D	m
^N	14	0E	♫	SO	46	2E	.	78	4E	N	110	6E	n
^O	15	0F	☼	SI	47	2F	/	79	4F	O	111	6F	o
^P	16	10	►	SLE	48	30	0	80	50	P	112	70	p
^Q	17	11	◄	CS1	49	31	1	81	51	Q	113	71	q
^R	18	12	↕	DC2	50	32	2	82	52	R	114	72	r
^S	19	13	‼	DC3	51	33	3	83	53	S	115	73	s
^T	20	14	¶	DC4	52	34	4	84	54	T	116	74	t
^U	21	15	§	NAK	53	35	5	85	55	U	117	75	u
^V	22	16	▬	SYN	54	36	6	86	56	V	118	76	v
^W	23	17	↨	ETB	55	37	7	87	57	W	119	77	w
^X	24	18	↑	CAN	56	38	8	88	58	X	120	78	x
^Y	25	19	↓	EM	57	39	9	89	59	Y	121	79	y
^Z	26	1A	→	SIB	58	3A	:	90	5A	Z	122	7A	z
^[	27	1B	←	ESC	59	3B	;	91	5B	[	123	7B	{
^\	28	1C	∟	FS	60	3C	<	92	5C	\	124	7C	|
^]	29	1D	↔	GS	61	3D	=	93	5D	]	125	7D	}
^^	30	1E	▲	RS	62	3E	>	94	5E	^	126	7E	~
^_	31	1F	▼	US	63	3F	?	95	5F	_	127	7F	⌂†

† ASCII code 127 has the code DEL. Under MS-DOS, this code has the same effect as ASCII 8 (BS). The DEL code can be generated by the CTRL + BKSP key.