

TECHBOOKSBD

PROGRAMMING WITH C

COLLECTION

Zubayer-Al-Mahmud

2012

ZHRIDAY@GMAIL.COM

প্রগ্রামিং সি

1.01- সি প্রোগ্রামিং ল্যাংগুয়েজ টিউটোরিয়াল-০১

সি প্রোগ্রামিং ল্যাংগুয়েজ আমার শেখা প্রথম প্রোগ্রামিং ল্যাংগুয়েজ। এটি একটু অস্বাদ লাগলেও পরবর্তি যে কোন ল্যাং শিখতে সি এর জুরি নাই।

সি প্রোগ্রামিং ল্যাংগুয়েজ এর বৈশিষ্ট্য:

১. সি ল্যাংগুয়েজ অনেক বেশি পরিমাণ ফাংশন আছে এবং এর কম্পাইলার অ্যাসেম্বলি ল্যাংগুয়েজ সাপোর্ট করে। তাই যে কোন প্রোগ্রামই সি ভাষায় তৈরি করা সম্ভব।
২. এই ল্যাংগুয়েজ অনেক দ্রুত গতিতে কাজ করে। উদাহরণ স্বরূপ: ০ থেকে ১৫০০০ পর্যন্ত বৃদ্ধির প্রোগ্রামে সি ল্যাংগুয়েজ এ ১ সেকেন্ড সময় লাগলে বেশিকে লাগবে ৫০ সেকেন্ড।
৩. এ ভাষায় লেখা প্রোগ্রাম যেকোন কম্পিউটারে পরিবর্তন ছাড়া বা সামান্য পরিবর্তন করে চালানো যায়।
৪. স্ট্যাকচার্ড প্রোগ্রামিং ল্যাং হওয়ায় (ছোট ছোট ফাংশনে ভাগ করে) খুব সহজেই প্রোগ্রাম লেখা যায়।
৫. আরেকটি মজার বৈশিষ্ট্য হলো নিজের তৈরি ফাংশন সি লাইব্রেরীতে যোগকরে পরে তা ব্যবহার করা যায়।

সি ভাষার সহজ উদাহরণ:

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

```
int main()  
{  
printf(“I am alive! Beware.n” );  
getchar();  
return 0;  
}
```

ব্যাখ্যা:

১. #include
এটির মাধ্যমে একটি লাইব্রেরী ফাংশন stdio.h কে ডাকা হয় যেখানে অন্যান্য ফাংশন সম্পর্কে তথ্য রয়েছে।
২. main(){....}
এটি প্রগ্রামের প্রধান ফাংশন, এখান থেকেই প্রগ্রাম চালু হয়।
৩. printf(“I am alive! Beware.n”);
এটি I am alive! Beware. বক্যটি স্ক্রিনে প্রদর্শন করায়।
৪. n দিয়ে নতুন লাইন বুঝানো হয়েছে।
৫. getchar();
একটি কী প্রেস না করা পর্যন্ত আউটপুট প্রদর্শন করতে থাকে।

1.02- টারবো সি কম্পিউটারে সেট আপ করা সি প্রগ্রামিং ল্যাংগুয়েজ টিউটরিয়াল-০২



[পূর্বের পার্টে](#) মূলত: সি প্রগ্রামের প্রাথমিক আলোচনা করেছি। এখন

দেখবো কিভাবে টারবো সি প্রগ্রাম সেট আপ ও প্রগ্রাম লিখে কম্পাইল ও রান করা যায়।

টারবো সি প্রগ্রাম ডাউনলোড করা ও সেট আপ করা

[এই লিংক](#) থেকে টারবো সি ডাউনলোড করুন। দেখুন ৪.১৭ মেগাবাইটের ফাইলটি ডাউনলোড হয়েছে কিনা।

Tc কে আনজিপ করুন। আপনার কম্পিউটারের সি ড্রাইভে tc নামের ফোল্ডারে সেট আপ হবে।

এবার c: ড্রাইভের tc ফোল্ডারের bin নামের ফোল্ডারটি খুলুন। এখানে tc.exe ফাইলটিই মূল c প্রগ্রাম, tc.exe তে ডাবল ক্লিক করুন।
পাসের ছবির মতো আসে কিনা দেখুন।

প্রগ্রাম লিখে কম্পাইল ও রান করা

প্রথম পার্টের কোডগুলো লিখুন।

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

```
int main()  
{  
    printf("I am alive! Beware.n");  
    getch();  
    return 0;  
}
```

এবার Alt+F9 চাপুন। আপনার প্রগ্রামে কোন ভুল থাকলে তার বর্ণনা নিচের দিকে দেখতে পাবেন; সংশোধন করে Alt+F9 চেপে দেখুন ভুল আছে কিনা, না থাকলে ctrl+F9 চাপুন।

স্ক্রিনে প্রদর্শিত হবে I am alive! Beware.

ctrl+F9 চাপার পর সি এর **কম্পাইলার** আপনার লিখিত কোডগুলোকে মেশিন কোডে পরিনত করে। এই ঘটনাটাকে বলি কম্পাইল করা।

*কম্পাইলার কি?

কম্পাইলার এক ধরনের কম্পিউটার প্রোগ্রাম বা প্রোগ্রাম-সমষ্টি যা কোন কম্পিউটার ভাষা (উঁস ভাষা) থেকে অপর একটি কম্পিউটার ভাষায় (গন্তব্য ভাষা) টেক্সট অনুবাদ করে। সাধারণত কোন প্রোগ্রামের সোর্সকোড থেকে মেশিনকোডে রূপান্তরের কাজটি কম্পাইলার দিয়ে করা হয়ে থাকে। আরো নিখুঁতভাবে বলতে গেলে, কম্পাইলার সোর্সকোডকে অ্যাসেম্বলি কোডে পরিনত করে, পরবর্তীতে অ্যাসেম্বলার অ্যাসেম্বলি কোডকে মেশিনকোডে এ পরিনত করে। কম্পিউটারের মাইক্রোপ্রসেসর এই মেশিনকোড বুঝতে পারে এবং সেই অনুযায়ী কর্মসম্পাদন করে।*

.. সূত্র: <http://bn.wikipedia.org/wiki/>

2.01- C টিউটোরিয়াল-৩(ক) টোকেন

এ পর্বে আলোচনার বিষয়:

১.টোকেন

২.কী ওয়ার্ড ও আইডেন্টিফায়ার

৩. কনস্ট্যান্ট

এই অধ্যায়টি অনেকটা ইংরেজী parts of speech এর মতো। তাই যারা সি শিখতে চান তারা মনোযোগ দিন।

সি টোকেন

প্রগ্রামিং ভাষায় ব্যবহার করা অর্থবোধক শব্দাংশকে টোকেন বলে।

টোকেনগুলো কয়েকটি ভাগে ভাগ করা হয়-

1. Keyword
2. Identifier
3. Constant
4. Operator
- 5.Special Symbol

1. Keyword: সব কী ওয়ার্ডএরই সুনির্দিষ্ট অর্থ আছে। এগুলো সি প্রগ্রামের নিজস্ব। নিচের টেবিলে কীওয়ার্ডগুলো দেয়া হলো-

- | | | | |
|------------|----------|------------|------------|
| • auto | • double | • int | • struct |
| • break | • else | • long | • switch |
| • case | • enum | • register | • typedef |
| • char | • extern | • return | • union |
| • const | • float | • short | • unsigned |
| • continue | • for | • signed | • void |
| • default | • goto | • sizeof | • volatile |
| • do | • if | • static | • while |

২. Identifier: প্রগ্রামারের দেয়া নাম হলো Identifier. যাচ্ছে তাই নাম দিলেই হবে না কিন্তু। নাম দিতে কিছু নিয়ম নীতি মেনে চলতে হবে। যেমন-

১. ছোট বা বড় যেকোন অক্ষর দিতে পারেন, সমস্যা নেই।
২. আন্ডারস্কোর দিতে পারেন তবে নামের মাঝে space দিতে পারবেন না।
৩. প্রথম অক্ষর সংখ্যা দেয়া যাবে না।
৪. Keyword এর নাম কোন Identifier কে দেয়া যাবে না।

2.02- সি টিউটোরিয়াল পার্ট-৩(খ)

এ পর্বে আলোচনা করবো: কনস্ট্যান্ট ও ভেরিয়েবল নিয়ে

কনস্ট্যান্ট:

প্রগ্রাম চলাকালীন যাদের মান পরিবর্তন করা যায়না তারাই কনস্ট্যান্ট। দুই রকমের কনস্ট্যান্ট আছে-

১. নিউমেরিক

ক. **integer:** দশমিক বিহীন সংখ্যা যেমন: ১২৩, -৩২১, ০, ৬৫৪৩২, +৭৮ ইত্যাদি। ইন্টিজার লিখতে স্পেস , কমা ইত্যাদি ব্যবহার করা যাবে না।

খ. **real:** দশমিক সহকারে ভগ্নাংশ সংখ্যা। যেমন-০.০০৮৩, -০. ইত্যাদি।

২. কারেক্টার

ক. **single:** যেমন: 'x', '5', '.' ইত্যাদি।

খ. **string:** যেমন: "Hello", "787" ইত্যাদি।

ভেরিয়াবল:

variable একটি পাত্রের মতো আচরণ করে। যখন যা রাখা হয়, ই তার মান হয়। মনে করুন, একটি পাত্রে দুধ রাখলে তাকে দুধের পাত্র বলি, সেই পাত্রেতে পানি রাখলে তা হবে পানির পাত্র।

প্রগ্রাম চলাকালীন সময়ে এর মানের পরিবর্তন হয়।

ভেরিয়াবলএর নাম দেয়ার ক্ষেত্রেও সুইনির্দিষ্ট নিয়ম আছে-

১. নাম ইংরেজী বর্ণ দিয়ে শুরু হবে।
২. আন্ডারস্কোর দেয়া যাবে, স্পেস দেয়া যাবে না।
৩. ইংরেজী বড়/ ছোট বা বড়-ছোট মিশানো হাতের লেখাও চলবে।

যেমন:

Average, height, Counter_1, TutorialBD ইত্যাদি।

3.0-সি প্রগ্রামে গ্রাফিক্স

যারা সি প্রগ্রামটি সেটআপ করেন নি তারা [এখান](#) থেকে শিখে নিতে পারেন। প্রকৃত পক্ষে সিস্টেম লেভেলের কাজ করতে এখনো সি এর জুরি নেই।

হার্ডওয়ারের আর পোর্টগুলো নিয়ে মজার মজার প্রগ্রাম নিয়ে হাজার হাজার অপেক্ষায় রইলাম।

নিচে একটি ছোট প্রগ্রাম দেখুন যা দিয়ে একটি বৃত্ত আঁকা হয়েছে।

```
#include<graphics.h>
#include<conio.h>

void main()
{
    int gd=DETECT, gm;

    initgraph(&gd, &gm, "c:\\turbo3\\bgi ");
    circle(200,100,150);

    getch();
    closegraph();
}
```

ব্যাখ্যা:

১. গ্রাফিক্সে কাজ করতে গেলে graphics.h লাইব্রেরীর দরকার হয়।

২. int gd=DETECT, gm;
initgraph(&gd, &gm, "c:\\turbo3\\bgi ");

ভিজিআই ডিভাইজ ডিটেক্ট করার জন্য ব্যবহৃত কোড। c:\\turbo3\\bgi পাথটি পরিবর্তিতও হতে পারে। আপনার সি প্রগ্রামের bgi ফোল্ডারের পাথটি ভিন্নও হতে পারে।

৩. circle(200,100,150);

কেন্দ্রবিন্দুর এক্স ও ওয়াই এর মান যথাক্রমে ২০০ ও

৪. closegraph() প্রগ্রামটিকে গ্রাফিক্স মুড থেকে টেক্সট মুডে ফেরত আনে।

এখন আরও একটি প্রগ্রাম দেখবো যা বেসিক শেপগুলো আঁকা হবে

```
/*
shapes.c
example 1.1
*/

#include<graphics.h>
#include<conio.h>

void main()
{
    int gd=DETECT, gm;
    int poly[12]={350,450, 350,410, 430,400, 350,350, 300,430, 350,450 };
    initgraph(&gd, &gm, "");
```

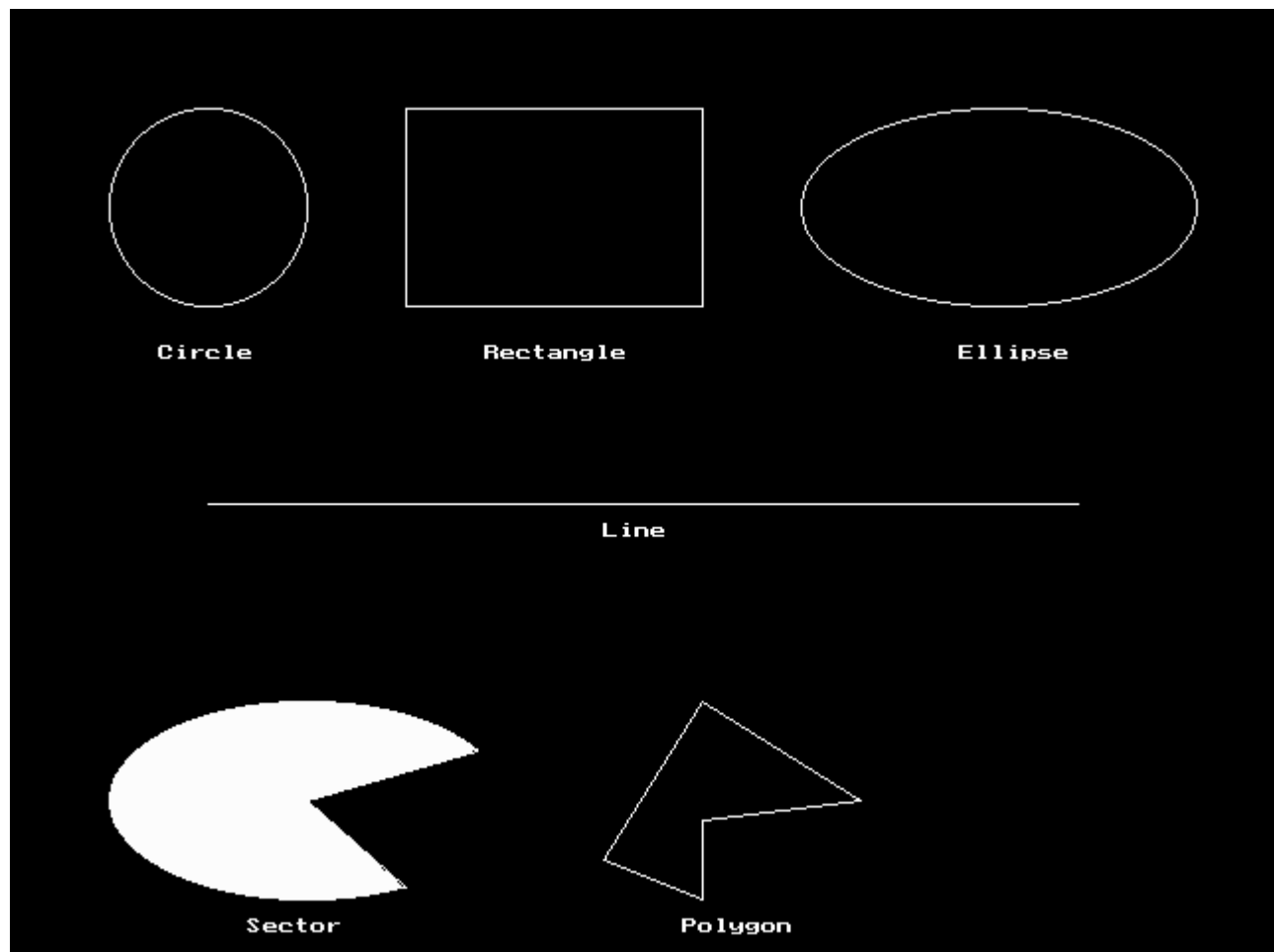
```

circle(100,100,50);
outtextxy(75,170, "Circle");
rectangle(200,50,350,150);
outtextxy(240, 170, "Rectangle");
ellipse(500, 100,0,360, 100,50);
outtextxy(480, 170, "Ellipse");
line(100,250,540,250);
outtextxy(300,260,"Line");

sector(150, 400, 30, 300, 100,50);
outtextxy(120, 460, "Sector");
drawpoly(6, poly);
outtextxy(340, 460, "Polygon");
getch();
closegraph();
}

```

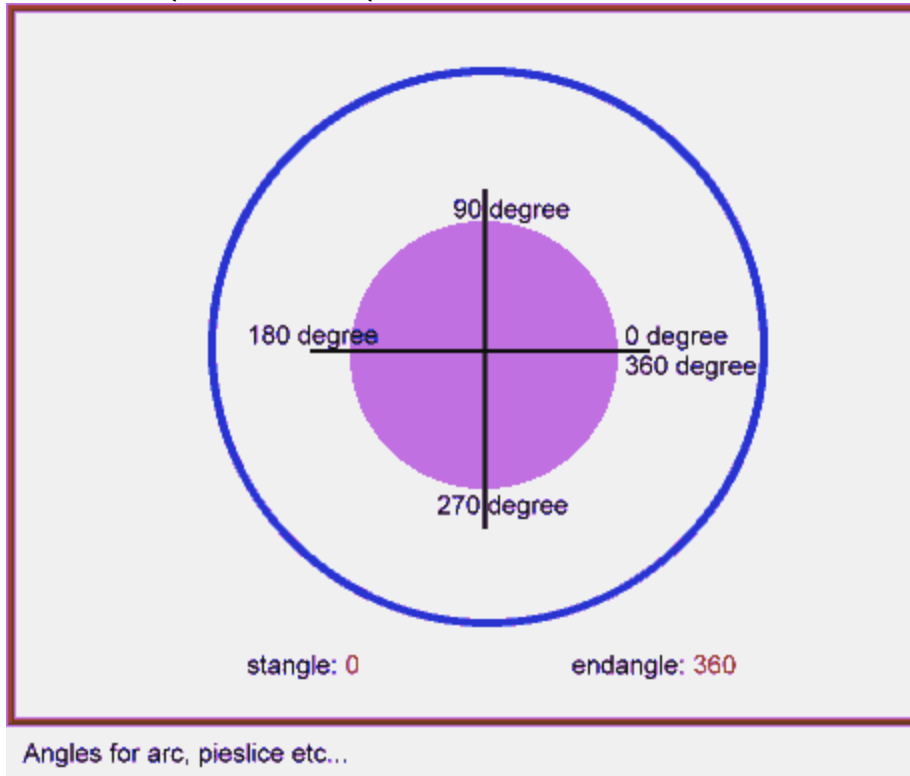
আউটপুট হবে:



ব্যখ্যা:

১. `outtextxy()` নির্দিষ্ট স্থানে কোন টেক্স লিখতে ব্যবহৃত হয়।
২. `arc(int x স্থানাঙ্ক, int y স্থানাঙ্ক, int শুরুর কোনের মান, int শেষের কোনের মান, int ব্যাসার্ধ);`
৩. `pieslice(int x, int y, int শুরুর কোনের মান, int শেষের কোনের মান, int ব্যাসার্ধ);`

কোনের ব্যাপারটা বুঝতে নিচের ছবিটি দেখুন



রঙের ব্যবহার:

এ ছাড়াও ভিন্ন ভিন্ন রঙের ব্যবহার করে আরও সুন্দর করা যায় গ্রাফিক্সকে। এর জন্য `setcolor(int রং);` ফাংশনটি ব্যবহার করতে হবে। ভিন্ন ভিন্ন রঙের জন্য ভিন্ন ভিন্ন মান ব্যবহার করতে হয়

BLACK: 0
BLUE: 1
GREEN: 2
CYAN: 3
RED: 4
MAGENTA: 5
BROWN: 6
LIGHTGRAY: 7
DARKGRAY: 8
LIGHTBLUE: 9
LIGHTGREEN: 10
LIGHTCYAN: 11
LIGHTRED: 12
LIGHTMAGENTA: 13
YELLOW: 14

WHITE: 15


এ বিষয়ে পরবর্তী পাঠে বিস্তারিত আলোচনা করা হবে। সাথে থাকুন। ধন্যবাদ।

<http://electrosofts.com/cgraphics/index.html>

প্রকাশ তারিখ: জানুয়ারি 18, 2010 সময় 2:51 অপরাহ্ন বিভাগ: [প্রগ্রামিং সি](#)

শেয়ার করুন¹⁰



লেখক: [মাহবুব টিউটো](#) |  লেখকের আর এস এস | <http://tutorialbd.com/bn>

তিনি টিউটোরিয়ালবিডির লেখক ও সম্পাদক। ১৯৮৩ সালে জন্ম। পড়ালেখার করেন ঢাকায়। ২০০৫ সালে স্নাতক পাস করেন। তিনি বেশ কিছু বাংলা ও ইংরেজী ব্লগের সাথে জড়িত আছেন। বর্তমানে তিনি [কনকর্ড গ্রুপে](#) কর্মরত আছেন। [ফেসবুকে](#) আর সাথে যোগাযোগ করতে পারেন।

তিনি আমাদের জন্য 296টি ব্লগ লেখেন এবং 582টি মন্তব্য প্রদান করেন। [মাহবুব টিউটো](#) এর পাতায় আপনারা আপনার আমন্ত্রণ।

2 টি মতামত | [সি প্রগ্রামে গ্রাফিক্স](#)

1.



[মাহামুদ](#) on অগাস্ট 11, 2011 at 1:51 পূর্বাহ্ন

খুব সুন্দর অনেক দিন পর আবার C নিয়ে বসলাম আপনার সাইট দেখে। এবার মনে হচ্ছে প্রগ্রাম শিখতে পারব। আপনি শুধু হেল্প করেন।
Pleas keep going.

[জবাব](#)

2.



[Sayed Ali Hasan](#) on ফেব্রুয়ারি 9, 2011 at 11:55 পূর্বাহ্ন

এই বিষয়ে গভীর জ্ঞান অর্জন দরকার। তাই আপনারা এই প্রচেষ্টা অব্যাহত রাখলে আমরা উপকৃত হবো। আপনারা এই হাতে থড়ি আমাদের পথ চলার প্রেরণা। আমাদের এই প্রেরণার পথ হঠাৎ করেই যেন বন্ধ না হয়ে যায়।

মোবাইল নং ০১৯১৩২৩৬৭৬২

ই-মেইল sayedalihasan@yahoo.com

4.0- মিউজিক্যাল কীবোর্ড বানানোর সহজ সি প্রজেক্ট

আগেই বলেছিলাম সি তে খুব সহজেই হার্ডওয়ার ডিভাইজ নিয়ে কাজ করা যায়। আমরা এ পর্বে সি প্রগ্রামিং এর মাধ্যমে কী বোর্ড নিয়ে কাজ করবো। এর পর মিউজিক্যাল কীবোর্ড বানানোর সহজ সি প্রজেক্টটি দেখবো।

প্রতিটি কী এর ASCII () মান রয়েছে। নিচের টেবিলে এই মানের ছক দেয়া হলো।

NON PRINTABLE CHARACTERS								
DEC	HEX	CHARACTER (CODE)	DEC	HEX	CHARACTER (CODE)			
0	0	NULL	16	10	DATA LINK ESCAPE (DLE)			
1	1	START OF HEADING (SOH)	17	11	DEVICE CONTROL 1 (DC1)			
2	2	START OF TEXT (STX)	18	12	DEVICE CONTROL 2 (DC2)			
3	3	END OF TEXT (ETX)	19	13	DEVICE CONTROL 3 (DC3)			
4	4	END OF TRANSMISSION (EOT)	20	14	DEVICE CONTROL 4 (DC4)			
5	5	END OF QUERY (ENQ)	21	15	NEGATIVE ACKNOWLEDGEMENT (NAK)			
6	6	ACKNOWLEDGE (ACK)	22	16	SYNCHRONIZE (SYN)			
7	7	BEEP (BEL)	23	17	END OF TRANSMISSION BLOCK (ETB)			
8	8	BACKSPACE (BS)	24	18	CANCEL (CAN)			
9	9	HORIZONTAL TAB (HT)	25	19	END OF MEDIUM (EM)			
10	A	LINE FEED (LF)	26	1A	SUBSTITUTE (SUB)			
11	B	VERTICAL TAB (VT)	27	1B	ESCAPE (ESC)			
12	C	FF (FORM FEED)	28	1C	FILE SEPARATOR (FS) RIGHT ARROW			
13	D	CR (CARRIAGE RETURN)	29	1D	GROUP SEPARATOR (GS) LEFT ARROW			
14	E	SO (SHIFT OUT)	30	1E	RECORD SEPARATOR (RS) UP ARROW			
15	F	SI (SHIFT IN)	31	1F	UNIT SEPARATOR (US) DOWN ARROW			
PRINTABLE CHARACTERS								
DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER
32	0x20	<SPACE>	64	0x40	@	96	0x60	`
33	0x21	!	65	0x41	A	97	0x61	a
34	0x22	“	66	0x42	B	98	0x62	b
35	0x23	#	67	0x43	C	99	0x63	c
36	0x24	\$	68	0x44	D	100	0x64	d
37	0x25	%	69	0x45	E	101	0x65	e
38	0x26	&	70	0x46	F	102	0x66	f
39	0x27	‘	71	0x47	G	103	0x67	g
40	0x28	(72	0x48	H	104	0x68	h
41	0x29)	73	0x49	I	105	0x69	i
42	0x2A	*	74	0x4A	J	106	0x6A	j
43	0x2B	+	75	0x4B	K	107	0x6B	k
44	0x2C	,	76	0x4C	L	108	0x6C	l
45	0x2D	-	77	0x4D	M	109	0x6D	m
46	0x2E	.	78	0x4E	N	110	0x6E	n
47	0x2F	/	79	0x4F	O	111	0x6F	o
48	0x30	0	80	0x50	P	112	0x70	p
49	0x31	1	81	0x51	Q	113	0x71	q
50	0x32	2	82	0x52	R	114	0x72	r
51	0x33	3	83	0x53	S	115	0x73	s
52	0x34	4	84	0x54	T	116	0x74	t

53	0x35	5	85	0x55	U	117	0x75	u
54	0x36	6	86	0x56	V	118	0x76	v
55	0x37	7	87	0x57	W	119	0x77	w
56	0x38	8	88	0x58	X	120	0x78	x
57	0x39	9	89	0x59	Y	121	0x79	y
58	0x3A	:	90	0x5A	Z	122	0x7A	z
59	0x3B	;	91	0x5B	[123	0x7B	{
60	0x3C	<	92	0x5C	\	124	0x7C	
61	0x3D	=	93	0x5D]	125	0x7D	}
62	0x3E	>	94	0x5E	^	126	0x7E	~
63	0x3F	?	95	0x5F	_	127	0x7F	

EXTENDED ASCII CHARACTERS

DEC	HEX	CHARACTER	DEC	HEX	CHARACTER	DEC	HEX	CHARACTER
128	0x80	€	171	0xAB	«	214	0xD6	Ö
129	0x81	◆	172	0xAC	¬	215	0xD7	×
130	0x82	,	173	0xAD		216	0xD8	Ø
131	0x83	f	174	0xAE	®	217	0xD9	Ù
132	0x84	„	175	0xAF	-	218	0xDA	Ú
133	0x85	...	176	0xB0	°	219	0xDB	Û
134	0x86	†	177	0xB1	±	220	0xDC	Ü
135	0x87	‡	178	0xB2	²	221	0xDD	Ý
136	0x88	^	179	0xB3	³	222	0xDE	Þ
137	0x89	‰	180	0xB4	´	223	0xDF	ß
138	0x8A	Š	181	0xB5	µ	224	0xE0	à
139	0x8B	‹	182	0xB6	¶	225	0xE1	á
140	0x8C	Œ	183	0xB7	·	226	0xE2	â
141	0x8D	◆	184	0xB8	¸	227	0xE3	ã
142	0x8E	Ž	185	0xB9	¹	228	0xE4	ä
143	0x8F	◆	186	0xBA	º	229	0xE5	å
144	0x90	◆	187	0xBB	»	230	0xE6	æ
145	0x91	‘	188	0xBC	¼	231	0xE7	ç
146	0x92	’	189	0xBD	½	232	0xE8	è
147	0x93	“	190	0xBE	¾	233	0xE9	é
148	0x94	”	191	0xBF	¿	234	0xEA	ê
149	0x95	•	192	0xC0	À	235	0xEB	ë
150	0x96	—	193	0xC1	Á	236	0xEC	ì
151	0x97	—	194	0xC2	Â	237	0xED	í
152	0x98	~	195	0xC3	Ã	238	0xEE	î
153	0x99	™	196	0xC4	Ä	239	0xEF	ï
154	0x9A	š	197	0xC5	Å	240	0xF0	ð
155	0x9B	›	198	0xC6	Æ	241	0xF1	ñ
156	0x9C	œ	199	0xC7	Ç	242	0xF2	ò
157	0x9D	◆	200	0xC8	È	243	0xF3	ó
158	0x9E	ž	201	0xC9	É	244	0xF4	ô
159	0x9F	ÿ	202	0xCA	Ê	245	0xF5	õ
160	0xA0		203	0xCB	Ë	246	0xF6	ö
161	0xA1	¡	204	0xCC	Ì	247	0xF7	÷
162	0xA2	¢	205	0xCD	Í	248	0xF8	ø
163	0xA3	£	206	0xCE	Î	249	0xF9	ù

164	0xA4	¤	207	0xCF	İ	250	0xFA	ú
165	0xA5	¥	208	0xD0	Ð	251	0xFB	û
166	0xA6	!	209	0xD1	Ñ	252	0xFC	ü
167	0xA7	§	210	0xD2	Ò	253	0xFD	ý
168	0xA8	¨	211	0xD3	Ó	254	0xFE	þ
169	0xA9	©	212	0xD4	Ô	255	0xFF	ÿ
170	0xAA	ª	213	0xD5	Õ			

এ মানের উপর ভিত্তি করে কিভাবে কীবোর্ড ইভেন্টে কাজ করা যায় তাও দেখবো। এর পর হয়তো কোন প্রজেক্টে মাউস ডিটেক্ট এবং মাউস ইভেন্ট আরও কিছু দেখতে পার।

এবার দেখি মিউজিক্যাল কীবোর্ড বানানোর সহজ প্রজেক্ট

```

1. #include
2. #include
3. #include
4. void main()
5. {
6. char ch='y';
7. clrscr();
8. printf("Press X to exit.....");
9. while(ch !='X')
10. {
11. ch = getch();
12. sound(20*ch);
13. delay(75);
14. nosound();
15. }
16. }
```

ব্যাখ্যা:

লাইন 1-3:

প্রথম তিন লাইন হেডার ফাইল কল করা হয়েছে।

লাইন-4:

“void main()” প্রধান ফাংশন যা এক্সিকিউট হয়।

লাইন-5:

ব্রাকেট যেখান থেকে প্রগ্রাম শুরু হয়।

লাইন-6:

character ভেরিয়ার্স ডিক্লার করা হয় এবং তার মান 'y' দেয়া হয়।

লাইন-7:

স্ক্রিনে আগের কোন কারেক্টার থাকলে তা পরিষ্কার করা হয়।

লাইন-8:

ইউজারকে বলা হয় যে আপনি X চাপলে বন্ধ হবে।

লাইন-9:

while loop শুরু। এর মধ্যে ইনস্ট্রাকশনগুলো বারবার (প্রতি কী প্রেসে) চালু হয়।

লাইন-10:

while loop শুরু।

লাইন-11:

এই লাইনের মাধ্যমে প্রতি কী প্রেসে ch ভ্যারিয়ার্সের ASCII মান সংরক্ষিত হয়।

লাইন-12:

“sound()” এর মাধ্যমে পিসি স্পিকার ASCII মান এর ২০ গুন মানের ফ্রিকুয়েন্সির শব্দ তৈরী করে। বিভিন্ন কী প্রেসে বিভিন্ন ফ্রিকুয়েন্সির শব্দ তৈরী হয়।

লাইন-13:

“delay()” এর মাধ্যমে শব্দ কতক্ষণ স্থায়ী হবে তা বুঝায়।delay(75) এ বলতে বুঝায় শব্দটি ৭৫ মিলিসেকেন্ড স্থায়ী হবে।

লাইন-14:

“nosound()” ফাংশনটি “sound()” এর বিপরিত। এটি মূলত স্পিকার বন্ধ করে দেয়।

লাইন-15:

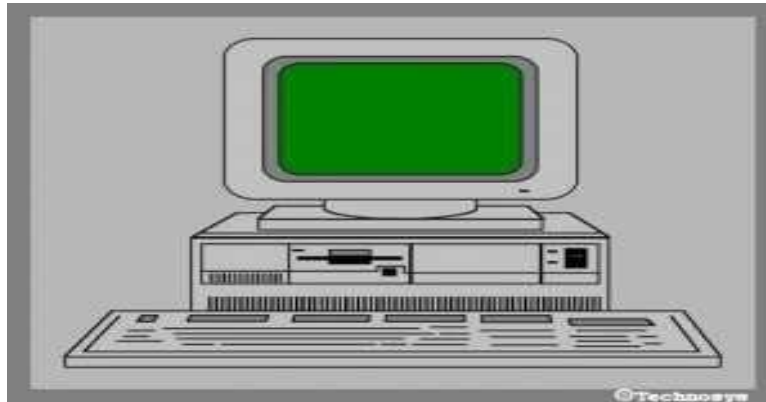
while loop শেষ।

লাইন-16:

“main()” ফাংশন শেষ।

আপনি এখান থেকে [কোড](#) ডাউনলোড করতে পারেন।

5.01- সি তে মজার মজার প্রোগ্রাম



মিড লেভেল ভাষা হিসেবে [সি](#) একটি জনপ্রিয় বহুল ব্যবহৃত ভাষা। উচ্চস্তরের ভাষা শিক্ষার সিঁড়ি হিসেবে শিক্ষা প্রতিষ্ঠান সমূহে সি শেখানো হয়। সকল প্রোগ্রামের ভিত্তিই হল সি। তাই বন্ধুরা, সি কে সহজভাবে উপস্থাপন করাই আমার মূল লক্ষ্য। তোমরা যদি আমার লেখা একটু মনঃযোগ দিয়ে পড় এবং একটু কষ্ট করে অনুশীলন কর তবে তোমরা নিজের মুখেই একথা স্বীকার করবে এবং অবাক হবে যে, সি এত সহজ! চলো বন্ধুরা এবার শুরু থেকেই শুরু করি।

আমরা এমনভাবে শুরু করব যেন সহজেই বুঝতে পারি কোন স্টেটমেন্ট কেন ব্যবহৃত হয়? নিচের প্রোগ্রামটি লক্ষ্য কর :

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
printf("This is my first c programming");
```

```
getch();
```

```
}
```

Output: প্রোগ্রামটি রান করলে নিম্নোক্ত ফলাফল পোয়া যাবে:

This is my first c programming

মূল বর্ণনা:

১) আমরা শুরুতেই দেখতে পাচ্ছি `#include<stdio.h>` এবং `#include<conio.h>` দিয়ে শুরু করা হয়েছে। এখানে `stdio.h` এর পূর্ণরূপ হল standard input output header file এবং `conio.h` এর পূর্ণরূপ হল control input output header file . এখন প্রশ্ন হল এগুলো কেন ব্যবহার করব ? আচ্ছা তাহলে একটা গল্প শোন। এর মধ্যেই উত্তরটি রয়েছে।

একজন লোক গরু কিনবে। এখন গরুর মালিকের কাছে গিয়ে লোকটি বলছে, ভাই আমার একটি গরু কিনতে হবে। এখন আপনি যদি আপনার গরুটি আমার কাছে বিক্রি করেন তাহলে আমার উপকার হত। গরুর মালিক গরুটি বিক্রি করার সিদ্ধান্ত নিয়ে বলল- ঠিক আছে আপনি ঐ মাঠে গিয়ে গরুটি দেখে আসুন, দেখুন আপনার পছন্দ হয় কিনা? লোকটি বলল- সেখানে তো অনেক গরু আছে। আমি চিনবো কিভাবে কোন গরুটা আপনার। গরুর মালিক বলল- আমার গরুর মাথায় সাদা কয়েকটি পশম আছে আর সারা শরীরের লোম কালো। লোকটি তাকে ধন্যবাদ জানিয়ে চলে গেল।

এখানে লোকটি হল কম্পাইলার, যে প্রোগ্রামটি পরীক্ষা করবে কোথাও ভুল আছে কিনা। `stdio.h` এবং `conio.h` হল গরুর মালিক। `stdio.h` বলছে `printf()` আমার ফাংশন এবং `conio.h` বলছে `getch()` আমার ফাংশন। সুতরাং বুঝতেই পারছ বন্ধুরা, `stdio.h` এবং `conio.h` কম্পাইলারকে পরিচয় করিয়ে দেয়ার জন্য ব্যবহৃত হচ্ছে। আর `#include<>` কে মনে কর একটা চেয়ার যেখানে বসে `stdio.h` এবং `conio.h` অর্ডার করছে।

২) `void main()` ব্যবহৃত হয় একারণে যে, এখান থেকে মূল প্রোগ্রাম শুরু হবে। এরপর একটা ‘ { ‘ চিহ্ন বসবে এবং প্রোগ্রাম শেষে ‘ } ‘ চিহ্ন দিয়ে শেষ হবে। `main()` এর পূর্বে `void` না দিলো হয় তবে ‘ } ‘ চিহ্নের পূর্বে `return` ; ফাংশন দিতে হয়।

৩) পূর্বেই বলেছি `printf()` একটা ফাংশন। এর শেষে ; চিহ্ন দিতে হয়। (” “)এর মধ্যে যা লেখা হবে output এ তাই পোয়া যাবে।

৪) `getch()` ফাংশনের মাধ্যমে output কে স্থির রাখা যাবে। এই ফাংশন ব্যবহার না করলে output দেখতে হবে window menu থেকে output সাবমেনুতে।

5.02– গল্পে গল্পে সি প্রোগ্রামিং

নিচের প্রোগ্রামটি একটু মনঃযোগ দিয়ে লক্ষ্য কর। প্রোগ্রামটির মাধ্যমে দুটি সংখ্যার মধ্যে যোগফল নির্ণয় করা যাবে।

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
int a;
```

```
int b;
```

```
int Result;
```

```
printf(“First Number:”);
```

```
scanf(“%d”,&a);
```

```
printf("Second Number:");

scanf("%d",&b);

Result=a+b;

printf("Result=%d",Result);

getch();

}
```

মূল বর্ণনা:

বন্ধুরা প্রোগ্রামটি দেখে হয়তো তোমাদের খুব ভয় করছে। আসলে ভয়ের কিছুই নেই। এটি অত্যন্ত সহজ একটি প্রোগ্রাম। চলো, এবার একটি মজার গল্পের মাধ্যমে প্রোগ্রামটি বোঝার চেষ্টা করি।

” মীমের বাসায় দু’জন লোক বেড়াতে এসেছে। তাদের মধ্যে একজন খুবই লম্বা আর একজন খাটো। লোকদুটো মীমের সাথে পরিচিত হল। একটি লোক তাকে জিজ্ঞাসা করল- আচ্ছা মীম, তুমি কি করতে পছন্দ কর? মীম বলল- আমি কোন ছবিতে রং করতে পছন্দ করি। লোকটি বলল- আচ্ছা, তুমি কোন ছবি অংকন করতে পছন্দ কর? মীম বলল- বাংলাদেশের জাতীয় পতাকা। লোকটি বলল- তুমি কি জানো, বাংলাদেশের জাতীয় পতাকা অংকন করতে কি কি রং লাগে? মীম বলল- হ্যা জানি। লোকটি বলল- তুমি একটা কাজ কর। রং দুটির নাম একটি খাতায় লিখে আমাকে দো। আমি ঐ দুটি রং কিনে তোমাকে উপহার দেব। মীম খুবই খুমশ হয়ে দ্রুত একটি খাতা নিয়ে এসে নিচের মত করে লিখল:

১. জাতীয় পতাকা অংকন করতে প্রথম রংটির নাম হল:

লাল

২. জাতীয় পতাকা অংকন করতে দ্বিতীয় রংটির নাম হল:

সবুজ

এরপর কাগজটি লোকটির হাতে দিয়ে মীম বলল-এই দুটি রং দিয়ে জাতীয় পতাকা অংকন করে একটি বাঁশের খুঁটির সাহায্যে বেঁধে উড়ানো যাবে।
—এই বলে সে ফিক করে হেসে ফেলল।”

গল্পটি নিশ্চয়ই মন:যোগ দিয়ে পড়েছ? এবার চলো দেখি এই গল্পের মাধ্যমে কিভাবে প্রোগ্রামটি বোঝা যায়?

১) বন্ধুরা, আমরা প্রোগ্রামটিতে প্রথমেই `int a; int b; int Result` ব্যবহার করেছি। এখানে `int` এর পূর্ণরূপ হল `integer` (পূর্ণসংখ্যা)। অর্থাৎ আমরা যে যোগটি করব তা পূর্ণসংখ্যা না দশমিক সংখ্যা হবে `int` তা নির্ধারণ করে। গল্পে লোকটি মীমকে বলেছে একটি জাতীয় পতাকার রং করতে কি কি লাগে? মীম বলেছে- লাল ও সবুজ রং এবং একটি বাঁশের খুঁটির সাথে বেঁধে দিলে পতাকাটি উড়বে। আমরাও তদ্রূপ বলছি, দুটি সংখ্যা যোগ করতে প্রথমেই দুইটি সংখ্যা নিতে হবে। অর্থাৎ গল্পে লাল রং বলতে `a` নামক একটি সংখ্যা, সবুজ রং বলতে `b` নামক একটি সংখ্যা এবং একটি বাঁশের খুঁটির খুঁটির সাহায্যে উড়বে বলতে বোঝানো হচ্ছে যে, `Result` এর মাধ্যমে আমরা আউটপুট দেখব।

২) গল্পে মীম লিখেছে- জাতীয় পতাকা অংকন করতে প্রথম রংটি হল:

এরপর নিচে লিখেছে লাল। আর আমরা এই জিনিসটি printf() এর মধ্যে লিখেছি “দুটি সংখ্যা যোগ করতে প্রথম সংখ্যাটি হল:” (“First Number”)। এরপর নিচে scanf() এর মধ্যে লিখেছি প্রথম সংখ্যা a। অর্থাৎ scanf(“%d",&a); স্টেটমেন্টটিতে %d লিখলাম এই কারণে যে, আমরা পূর্বে int ব্যবহার করেছি। যদি int এর পরিবর্তে float লিখতাম তবে %f লিখতাম। %d এর পরে &a এর অর্থ a সংখ্যার জন্য একটি input নিবে।

৩) গল্পে মীম এরপরই লিখেছে- জাতীয় পতাকা অংকন করতে দ্বিতীয় রংটি হল:

এরপর নিচে লিখেছে সবুজ। আমরা printf() এর মধ্যে লিখেছি (“Second Number:”) এরপর নিচে লিখেছি দ্বিতীয় সংখ্যা b অর্থাৎ scanf(“%d",&b) statement টির মাধ্যমে যদি প্রশ্ন করি “%d” কার জন্য? উত্তর হবে b এর জন্য কারণ “%d” এর পরে কমা দিয়ে &b লেখা আছে।

৪) এবার সংখ্যা দুটি যোগ করা হয়েছে Result=a+b; statement টির মাধ্যমে। সুতরাং a একটি সংখ্যা এবং b আরেকটি সংখ্যা যোগ করে যোগফল থাকবে Result এর মধ্যে। গল্পের মত লাল ও সবুজ রং দিয়ে পতাকা অংকন করে পতাকাটি উড়বে বাশের খুঁটির মাধ্যমে।

৫) এবার printf(“Result=%d”,Result); statement টির মাধ্যমে যোগফল পোয়া যাবে। আগেই বলেছি ডাবল কোটেশনের (“”) মধ্যে যা লেখা যায় আউটপুটে তাই দেখতে পোয়া যায়। সে হিসেবে আউটপুটে দেখার কথা Result=%d , কিন্তু তা হবে না। Result= এতটুকুই থাকবে শুধু %d থাকবে না। কম্পাইলারের কাছে এর অন্য একটি অর্থ আছে। অর্থাৎ %d কার জন্য? পাশেই লিখেছি Result এর জন্য। সুতরাং পূর্বের লাইনের যোগকৃত ফলাফল %d এর এখানে বসে থাকবে।

OUTPUT: প্রোগ্রামটি রান করলে নিম্নের আউটপুট পোয়া যাবে:

First Number:10

Second Number: 20

Result=30

উপরের প্রোগ্রামটি নিচের মত ছোট করে লিখা যাবে। এতে আউটপুটের যোগফলের কোন পরিবর্তন হবে না।

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
int a,b,Result;
```

```
printf(“First and Second Number:”);
```

```
scanf(“%d%d",&a,&b);
```

```
Result=a+b;
```

```
printf("Result=%d",Result);
```

```
getch();
```

```
}
```

সুতরাং প্রোগ্রাম দুটির মধ্যে পার্থক্য হচ্ছে প্রথম প্রোগ্রামে `int a; int b; int Result;` আলাদা লাইনে লেখা হয়েছে এবং `int` কথাটা তিনবার লেখা হয়েছে। কিন্তু এখানে একটি `int` দিয়ে `a,b,Result;` লেখা হয়েছে মাঝখানে শুধু কমা চিহ্ন দিয়ে। তদ্রূপ `printf()` এর মধ্যে লেখা হয়েছে `First and Second Number:`

একটি `scanf()` এর মধ্যেই দুটি সংখ্যা নেয়া হয়েছে `a` ও `b`। আর এভাবেই প্রোগ্রামটি ছোট হয়ে গেছে।

একটি মজার জিনিস:

উপরের প্রোগ্রামে দুটি সংখ্যা যদি ৩২৭৬৭ এর চেয়ে বেশী হয় তবে প্রোগ্রামের আউটপুট ভুল দেখাবে। সেক্ষেত্রে `int` এর পূর্বে `long` ব্যবহার করতে হবে। অর্থাৎ মীমের বাসায় যে দু'জন লোক এসেছিল এদের মধ্যে লম্বা লোকটি যত উপরের জিনিস ছুটে পারবে খাটো লোকটি কিন্তু অত উঁচু জিনিস ছুতে পারবে না। তাই খাটো লোকটি হল `int` আর লম্বা লোকটি হল `long int`। `int` এর সীমা `-32768` থেকে `+32768` এবং `long int` এর সীমা `-2147483648` `-2147483647` পর্যন্ত। আর `%d` এর জায়গায় দিতে হবে `%ld`।

আমি প্রোগ্রামের শুধু মূল অংশটুকুই বর্ণনা করেছি। সকল প্রোগ্রামের জন্য নিচের অংশটুকু মোটামুটিভাবে বাধ্যতামূলক।

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
getch();
```

```
}
```

5.03– গল্পের মাধ্যমে সি প্রোগ্রামিং শেখা

নিচের প্রোগ্রামটি লক্ষ্য করো:

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

```
#include<conio.h>
```

```
void main()
```

```
{  
  
int a;  
  
printf("Press 1 or 2 or 3 or 4:");  
  
scanf("%d",&a);  
  
switch(a)  
  
{  
  
case 1:  
  
printf("\nYou press one.");  
  
break;  
  
case 2:  
  
printf("\nYou press two.");  
  
break;  
  
case 3:  
  
printf("\nYou press three.");  
  
break;  
  
case 4:  
  
printf("\nYou press four.");  
  
break;  
  
default:  
  
printf("\nYou do not press 1 or 2 or 3 or 4");  
  
break;  
  
}  
  
getch();
```

}

মূল বর্ণনা:

প্রোগ্রামটি নি:শ্চয় মন:যোগ সহকারে লক্ষ্য করেছে। চলো বন্ধুরা দেখি, আজকে আমাদের জন্য কোন গল্পটা অপেক্ষা করছে? যে গল্পের মাধ্যমে আমরা প্রোগ্রামটি শিখতে পারব।

” ইম্পেক্টর পিন্টু তার কিছু সেন্সি নিয়ে একটি চোরের পেছনে ধোয়া করছে। একসময় তারা একটি চার রাস্তার মোরে এসে দাড়িয়ে পড়ল। তারা কেউই বুঝতে পারছে না যে, চোরটি কোন রাস্তা দিয়ে পালিয়েছে? ইম্পেক্টর পিন্টু তার একটি সেন্সি জিতকে জিজ্ঞাসা করল- চোরটিকে কি তুমি সত্যিই দেখেছিলে? জিত বলল- স্যার, আমি ভালো করে দিখতে পারি নি। তবে অন্ধকারে পায়ের শব্দ শুনে আমি মনে করেছি এটা চোরের পায়ের শব্দ। তাই আপনাকে দ্রুত সংবাদ দিয়ে চোরটির পেছনে ধোয়া করলাম।

ইম্পেক্টর পিন্টু বললেন- তার মানে এটি চোর কিনা তা নিশ্চিত নয়। হুম, ভাববার বিষয়। আচ্ছা, এটা যদি চোর হয় তবে কোন রাস্তা দিয়ে গেল, এটো তো বুঝতে পারছি না। যাইহোক, একটা কাজ করি। আমরা চারটি পয়েন্টে আগাতে পারি। এক: চোরটি যদি পূর্ব দিকের রাস্তা দিয়ে যায়, তবে ঐ দিকে কিছু দূরেই আমাদের ১নং ক্যাম্প আছে। সেখানে ফোন করে জানিয়ে দিই চোরটিকে ধরার জন্য। দুই: চোরটি যদি পশ্চিম দিকের রাস্তা দিয়ে যায় তবে ঐ দিকে আমাদের ২নং ক্যাম্প আছে। সেখানে জানিয়ে দিই। তিন: চোরটি যদি উত্তর দিকের রাস্তা দিয়ে যায় তবে ঐ দিকে আমাদের ৩নং ক্যাম্প আছে। সেখানে জানিয়ে দিই। চার: চোরটি যদি দক্ষিণ দিকের রাস্তা দিয়ে যায় তবে ঐ দিকে আমাদের ৪নং ক্যাম্প আছে। সেখানে জানিয়ে দিই। সুতরাং যেদিকেই যাক না কেন, চোরকে ধরা পড়তেই হবে। আরেকটি কথা, যদি কোন দিকেই চোরটি ধরা না পড়ে তবে এটা চোর নয়, এটা আমাদের সেন্সি জিত এর চোখের ধাঁধা। আর আমরা সবাই বোকা।”

গল্পটি খুবই মজার, তাই না বন্ধুরা?

১) আমরা `int a;` statement এর মাধ্যমে `a` নামক একটি পূর্ণসংখ্যা নিয়েছি। এরপর `printf()` ফাংশনের মধ্যে লিখেছি ”Press 1 or 2 or 3 or 4:” অর্থাৎ গল্পে বলা হয়েছে “চারটি পয়েন্টে আগাতে পারি।” সুতরাং আমাদেরকে যে কোন একটি সংখ্যা press করতে বলা হচ্ছে।

২) `scanf()` ফাংশনের মাধ্যমে আমরা `a` এর জন্য অর্থাৎ চোর ধরার জন্য input নিচ্ছি।

৩) `switch()` statement এর মাধ্যমে আমরা চারটি case দেখিয়েছি। গল্পে বলা হয়েছে, এক: চোরটি যদি পূর্ব দিকের রাস্তা দিয়ে যায় তবে ঐ দিকে আমাদের ১নং ক্যাম্প আছে। সেখানে ফোন করে জানিয়ে দিই চোরটিকে ধরার জন্য। আমরা বলছি case 1:

```
printf(“\nYou press one.”);
```

অর্থাৎ কীবোর্ড থেকে ১ চাপলে আউটপুটে দেখাবে You press one.

৪) গল্পে বলা হয়েছে, দুই: চোরটি যদি পশ্চিম দিকের রাস্তা দিয়ে যায় তবে ঐ দিকে আমাদের ২নং ক্যাম্প আছে। সেখানে ফোন করে জানিয়ে দিই চোরটিকে ধরার জন্য। আমরা বলছি case 2 :

```
printf(“\nYou press two.”);
```

অর্থাৎ কীবোর্ড থেকে ২ চাপলে আউটপুটে দেখাবে You press two.

৫) গল্পে বলা হয়েছে, তিন: চোরটি যদি উত্তর দিকের রাস্তা দিয়ে যায় তবে ঐ দিকে আমাদের ৩নং ক্যাম্প আছে। সেখানে ফোন করে জানিয়ে দিই চোরটিকে ধরার জন্য। আমরা বলছি case 3 :

```
printf("\nYou press three.");
```

অর্থাৎ কীবোর্ড থেকে ৩ চাপলে আউটপুটে দেখাবে You press three.

৬) গল্পে বলা হয়েছে, চার: চোরটি যদি দক্ষিণ দিকের রাস্তা দিয়ে যায় তবে ঐ দিকে আমাদের ৪নং ক্যাম্প আছে। সেখানে ফোন করে জানিয়ে দিই চোরটিকে ধরার জন্য। আমরা বলছি case 4 :

```
printf("\nYou press four.");
```

অর্থাৎ কীবোর্ড থেকে ৪ চাপলে আউটপুটে দেখাবে You press four.

৭) গল্পে শেষে বলা হয়েছে, যদি কোন দিকেই চোরটি ধরা না পড়ে তবে এটা চোর নয়। আমরা বলছি default:

```
printf("\nYou do not press 1 or 2 or 3 or 4");
```

অর্থাৎ যদি ১ কিংবা ২ কিংবা ৩ কিংবা ৪ এর কোনটিই press না করে অন্য কোন সংখ্যা press করি তবে output এ দেখাবে You do not press 1 or 2 or 3 or 4

৮) প্রত্যেকটি case এর শেষে break; দেয়া হয়েছে। কারণ ১ কিংবা ২ কিংবা ৩ কিংবা ৪ press করার পর যে কোন একটি case কাজ করবে। অর্থাৎ 1 চাপলে case 1 সম্পাদিত হবে। এরপর break; statement এর জন্য case 2 বা case 3 বা case 4 বা default এর কোনোটিই সম্পাদিত হবে না। আর switch() statement শুরুর জন্য “{” এবং শেষে “}” চিহ্ন দিতে হবে। switch(a) দেয়া হয়েছে এজন্য যে, a এর যে মান input হিসেবে নেয়া হবে তা case এর পরের সংখ্যার সাথে মিলে গেলে সেই case execution হবে। আর না মিললে default execution হবে। \n বলতে নতুন লাইন বোঝায়।

Output: প্রোগ্রামটি Run করলে নিম্নোক্ত Output পোয়া যাবে।

Press 1 or 2 or 3 or 4: 2

You press two.

পরিশেষ:

বন্ধুরা, আজ এ পর্যন্তই। তবে তোমরা কিন্তু আজকে switch() এর ব্যবহার শিখে গেলে। আজো আমি প্রোগ্রামের শুধু মূল অংশটুকুই বর্ণনা করেছি। সকল প্রোগ্রামের জন্য নিচের অংশটুকু মোটামুটিভাবে বাধ্যতামূলক।

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
getch();
```

```
}
```

তাই “{“ এবং ”}” অংশের মধ্যের অংশই বর্ণনা করেছে। তবে উপরের এই অংশ কেন বাধ্যতামূলক তা আমি পূর্বের “সি তে মজার মজার প্রোগ্রামিং” এ উল্লেখ করেছি। এরপর” গল্পে গল্পে সি প্রোগ্রামিং “এ আরো কিছু উল্লেখ করেছি।

6.01- ডাটা টাইপ: সি প্রোগ্রামিং টিউটোরিয়াল

আমরা যারা program লিখি তারা সবাই জানি C হচ্ছে programming শিখার হাতিয়ার। আসুন এ হাতিয়ার সম্পর্কে একটু ধারণা নেই। C প্রোগ্রামিং যে যে character গ্রহণ করে তা হল A-Z, a-z, + - * / = ! @ # \$ % ^ & * () } { _ / \ . , ? etc. অর্থাৎ যদি কোন প্রোগ্রাম লিখতে হয় উপরোক্ত character গুলো দিয়ে লিখতে হবে। আসুন এবার সর্বপ্রথম সচরাচর যে প্রোগ্রাম দিয়ে শুরু করা হয় তা দিয়েই শুরু করি।

```
#include <stdio.h>
library
main()

{
    printf("hello, world\n");
}
```

The first C program

?jakir!

hello, world

out put

- এটা খুবই ছোট একটা প্রোগ্রাম।
- এখানে #include দ্বারা প্রোগ্রামের head বুঝানো হয়েছে।
- stdio দ্বারা বুঝায় standard input output এবং .h দ্বারা header বুঝানো হয়।
- main() হচ্ছে একটি ফাংশন। এটি দ্বারা বুঝানো হয় যে প্রোগ্রামটি এখান থেকে শুরু হয়েছে। printf দ্বারা একটি command দেওয়া হয়েছে “Halo World” print করার জন্য। এখন যদি প্রোগ্রামটি run করাই তাহলে পর্দায় (Halo World) লেখাটি ফুটে উঠবে।

আজকে আমি সি প্রোগ্রামিং ল্যাংগুয়েজের ডাটা টাইপ সম্পর্কে আলচনা করব। অনেক প্রকারের ডাটা টাইপ আছে। তার মধ্য প্রধান চারটি হচ্ছে:

১. int data type
২. char data type
৩. float data type
৪. double data type

ইন্ডিজার:

Int data type বলতে integer quantity(অবিভাজ্য সংখ্যা যেমন: ১, ২, ৩ ইত্যাদি) বুঝায়। এর সাইজ ২ বাইট বা ১৬ বিট(১বাইট=৮বিট)এবং রেঞ্জ: -৩২৭৬৮ থেকে +৩২৭৬৭ পর্যন্ত।

কারেকটার:

char data type বলতে single character(একটি বর্ণ যেমন a, b, z, A, N ইত্যাদি) বুঝায়। এর সাইজ ১ বাইট বা ৮ বিট।বিট(১বাইট=৮বিট)এবং রেঞ্জ: -১২৮ থেকে +১২৭ পর্যন্ত।

ফ্লোট:

float data type বলতে floating point number(দশমিক সংখ্যা যেমন: ১০.৫, ১.৮, ৫.৬ ইত্যাদি) বুঝায়।এর সাইজ ৪ বাইট বা ৩২ বিট(১বাইট=৮বিট)এবং রেঞ্জ: ৩.৪ e-৩৮ থেকে ৩.৪ e+৩৮ পর্যন্ত।এখানে e দ্বারা exponent বা power বুঝানো হয়।

ডাবল:

double data type বলতে Double precision floating point number বুঝায়।এটা float data type এর মতোই তবে সাইজ বিশাল। এর সাইজ ৮ বাইট বা ৬৪ বিট।বিট(১বাইট=৮বিট)এবং রেঞ্জ: ১.৭ e-৩০৮ ১.৭ e+৩০৮ পর্যন্ত।

6.02- ডাটা টাইপ: সি প্রোগ্রামিং টিউটোরিয়াল

সব প্রোগ্রামই কিছু না কিছু প্রোগ্রাম নিয়ে কাজ করে। প্রোগ্রামে ব্যবহারের জন্য ডেটাকে প্রথমে মেমরিতে সংরক্ষণ করা হয় এবং প্রয়োজনে মেমরি থেকে ডেটা উত্তোলন করে কাজে লাগানো হয়। নিম্ন পদার্থের ভাষায় মেমরিতে ডেটা রাখার জন্য সরাসরি বিট, বাইট এবং মেমরি এ্যাদ্রেস ব্যবহার করা হয়, যা বড় বড় প্রোগ্রামের জন্য অত্যন্ত জটিল এবং কষ্টকর। কারণ লক্ষ লক্ষ এ্যাদ্রেসের মধ্যে কখন কোন এ্যাদ্রেস কোন ডেটা রাখা হল তা মনে রাখা অসম্ভব। এই অসুবিধা দূর করার জন্য এবং প্রোগ্রামকে সহজ করার লক্ষ্যে উচ্চ পদার্থের ভাষায় বিট বাইট এর পরিবর্তে ভেরিয়েবল ব্যবহার করা হয়।

ভেরিয়েবল:

ভেরিয়েবল হল মেমরির লোকেশনের নাম বা ঠিকানা। প্রোগ্রামে যখন কোন ডেটা নিয়ে কাজ করা হয়, প্রাথমিক ভাবে সেগুলো কমপিউটারের র্যামে অবস্থান করে। পরবর্তী সময়ে সেগুলো পুনরুদ্ধার বা পুনব্যবহারের জন্য ঐ নাম বা ঠিকানা জানা প্রয়োজন হয়। সুতরাং প্রোগ্রামে ডেটা নিয়ে কাজ করার সময় প্রতিটি ডেটার জন্য একটি ভেরিয়েবল ব্যবহার করাতে হয়। ভেরিয়েবল ব্যবহার না করে ও প্রোগ্রামে বিভিন্ন ধরনের ডেটা যেমন ক্যারেক্টার, স্ট্রিং, পূর্ণসংখ্যা, ভগ্নাংশ তথ্য দশমিক সংখ্যা সায়েন্টিফিকসংখ্যা তথ্য এক্সপোনেন্সিয়াল সংখ্যা নিয়ে কাজ করা যায়।

ডেটা টাইপ ও মডিফায়ার:

সি ভাষায় ভেরিয়েবল ব্যবহার করে ডেটা সংরক্ষণের জন্য ভেরিয়েবল ঘোষণার সময় ডেটার ধরন অনুযায়ী উপযুক্ত ডেটা টাইপ ও ঘোষণা করতে হয়। এরূপ ব্যবহৃত চারটি মৌলিক বা বিল্ট ইন ডেটা টাইপ হল: char, integer, float, double। চারটি মৌলিক ডেটা এর জন্য ঘোষিত ভেরিয়েবলের জন্য সংরক্ষিত মেমরি পরিসর, ডেটার প্রকৃতি এবং ধারণ ক্ষমতার একটা সীমাবদ্ধতা রয়েছে। যেমন: একটি char টাইপ ভেরিয়েবলের জন্য ৮ বিট বা ১ বাইট জায়গা সংরক্ষণ করে যাতে ০ থেকে ১২৭ মানবিশিষ্ট মোট ১২৮টি ব্যারেক্টরের যে কোন একটি মান রাখা যায়।

এছাড়াও সি তে আরও ৪টি মডিফায়ার আছে।

সেগুলো হলঃ- signed, unsigned, short, long ।

সাধারণত char টাইপ ভেরিয়েবলের জন্য signed ও unsigned মডিফায়ার এবং টাইপ ভেরিয়েবলের জন্য signed, unsigned, short ও long মডিফায়ার, float ও double টাইপ ভেরিয়েবলের জন্য short এবং long মডিফায়ার ব্যবহৃত হয়।

এসব ডেটার বিট ও মানের রেঞ্জ জানা খুব জরুরী।

ডেটা টাইপ	বিট সংখ্যা	ডেটা বা ভেরিয়েবলের মানের রেঞ্জ
Char	8	$-128 \leftrightarrow 127$ বা $-2^7 \leftrightarrow 2^7-1$
Signed char	8	$-128 \leftrightarrow 127$ বা $-2^7 \leftrightarrow 2^7-1$
Unsigned char	8	$0 \leftrightarrow 255$ বা $0 \leftrightarrow (2^8-1)$
Integer	16	$-32768 \leftrightarrow 32767$ বা $-2^{15} \leftrightarrow 2^{15}-1$
Short integer	16	$-32768 \leftrightarrow 32767$
Unsigned int	16	$0 \leftrightarrow 65535$ বা $0 \leftrightarrow (2^{16}-1)$
Long integer	32	$-2^{31} \leftrightarrow 2^{31}-1$
Float	32	$3.4 \times 10^{-38} \leftrightarrow 3.4 \times 10^{38}$
Long float	64	$1.7 \times 10^{-308} \leftrightarrow 1.7 \times 10^{308}$
Double	64	$1.7 \times 10^{-308} \leftrightarrow 1.7 \times 10^{308}$
Long double	80	$3.4 \times 10^{-4932} \leftrightarrow 1.1 \times 10^{4932}$

ভেরিয়েবল ঘোষনার নিয়মাবলী:

- একই ফাংশনে একই নামে দুই বা ততোধিক ভেরিয়েবল ঘোষণা করা যায় না।
- ভেরিয়েবল নামকরণে কেবল আলফাবেটিক ক্যারেক্টার (a .. z) (A...Z) ডিজিট (0,... ৯) এবং আন্ডারস্কোর(_) ও ডলার চিহ্ন (\$) ব্যবহার করা যায়। আন্ডারস্কোর ও ডলার চিহ্ন ব্যতীত অন্য কোন স্পেশাল ক্যারেক্টার (যেমন: !, +, -, % ইত্যাদি ব্যবহার করা যায় না)। যেমন: my_car, My\$Roll বৈধ, কিন্তু my@car , My&Roll অবৈধ।
- ভেরিয়েবল নামের মধ্যে কোন ফাকা স্থান থাকতে পারে না। যেমন: Myname, Myhouse বৈধ, My name, My house অবৈধ।
- ভেরিয়েবলের নাম ডিজিট বা অংক দিয়ে শুরু হতে পারে না।
- সি প্রোগ্রামে বড় হাতের অক্ষর এবং ছোট হাতের অক্ষর আলাদা অর্থ বহন করে। সি প্রোগ্রামে ছোট হাতের অক্ষর ব্যবহার করতে হয়। তবে বিশেষ ক্ষেত্রে বড় হাতের অক্ষর ব্যবহার করা হয়।
- কোন কীওয়ার্ডের নাম ভেরিয়েবল হিসেবে ব্যবহার করা যায় না।
- ভেরিয়েবল নামকরণে যে কোন সংখ্যক ক্যারেক্টার ব্যবহার করা যায়। তবে ভেরিয়েবলের নাম ৩১ ক্যারেক্টারের মধ্যে হওয়া ভাল।

7.0- চার ধরনের কনস্ট্যান্ট ও কয়েকটি প্রয়োজনীয় পরিভাষা: সি প্রোগ্রামিং টিউটোরিয়াল

চার ধরনের কনস্ট্যান্ট

ডাটা টাইপ এর মত C তে প্রধানত চার প্রকারের Constant বা ধ্রুবক আছে। (যাদের মান সব সময়ের জন্য স্থির তাদের ধ্রুবক বলে। যেমন ১ এর মান সবসময় ১ এ থাকবে) তারা হল:

1. Integer constant
2. Floating-point constant
3. Character constant
4. String constant

Integer এবং Floating-point constant সংখ্যা প্রকাশ করে। এদেরকে সাধারণত numeric-type constant ও বলা হয়। নিচের নিয়ম গুলো সকল numeric-type constant এর ক্ষেত্রে প্রযোজ্য।

- কমা এবং খালি স্পেস numeric-type constant এর ভিতর থাকতে পারবেনা।
- প্রয়োজন অনুযায়ী -(মাইনাস) সাইন ব্যবহার করা যাবে।
- numeric-type constant তাদের সর্বোচ্চ ও সর্বোচ্ছ নিম্ন সীমা অতিক্রম করতে পারবেনা।

Integer constant:

Integer constant বলতে integer quantity(অবিভাজ্য সংখ্যা যেমন: ১, ২, ৩ ইত্যাদি) বুঝায়। Integer constant তিন প্রকারের number system(সংখ্যা পদ্ধতি) এ লেখা হয়।

- **1. Decimal** (এদের ভিত্তি হচ্ছে **10**। সংখ্যা ০ হতে ৯ পর্যন্ত। অর্থাৎ ০ ১ ২ ৩ ৪ ৫ ৬ ৭ ৮ ৯)
- **2. Octal:** (এদের ভিত্তি হচ্ছে **৮**। সংখ্যা ০ হতে ৭ পর্যন্ত। অর্থাৎ ০ ১ ২ ৩ ৪ ৫ ৬ ৭)
- **3. Hexadecimal** ((এদের ভিত্তি হচ্ছে **16**। সংখ্যা ০ হতে ৯ এবং a b c d e f or A B C D E F পর্যন্ত। অর্থাৎ ০ ১ ২ ৩ ৪ ৫ ৬ ৭ ৮ ৯ a b c d e f or A B C D E F)

কম্পিউটার উপরের তিন সংখ্যা পদ্ধতির এক পদ্ধতি ও পড়তে পারে না। সে এগুলোকে **Binary** নামক অন্য সংখ্যা পদ্ধতিতে পরিবর্তন করে নেয়। Binary সংখ্যা পদ্ধতির ভিত্তি হচ্ছে ২ (০ এবং ১) ১ দ্বারা বিদ্যুতের উপস্থিতি এবং ০ দ্বারা বিদ্যুতের অন্তর্ভুক্তি প্রকাশ করে এবং কম্পিউটার এ ভাবেই কোন তথ্য পড়ে।

Floating-point constant:

দশমিক যুক্ত যেকোন সংখ্যাই হচ্ছে Floating-point constant। এগুলো সাধারণত Decimal সংখ্যা পদ্ধতিতেই লেখা হয়। যেমন: ১.২। ৫৮৪.৩। ১৫০.২১১। .০০০০৫৪ ইত্যাদি। Floating-point constant কে exponent ১০ এর power দ্বারা ও লেখা যায়।

Character constant:

Character constant বলতে single character বা একটি বর্ণকে বুঝায়। যা apostrophes (‘ ’) দ্বারা আবদ্ধ থাকে। যেমন: ‘a’ । ‘A’ । ‘c’ । ‘d’ ইত্যাদি। এগুলোর এক একটির এক একটি নির্দিষ্ট মান আছে। যা চিত্রে দেখতে পাবেন।

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(72	48	H	104	68	h
9	09	Horizontal tab	41	29)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

এখানে decimal ও Hexadecimal দুই মান ই দেওয়া আছে।

String constant:

String constant বলতে String কে বুঝায়। যা Double apostrophes (“ ”) দ্বারা আবদ্ধ থাকে। এগুলোর মান থাকে না। যেমন: “a” “mechi” “kaka” ইত্যাদি।

আমার টিউটোরিয়ালগুলো একটু নিরস হলেও C শেখার জন্য খুবই দরকারি। এগুলো হচ্ছে অ আ বা ক খ এর মত যা ছাড়া আপনি প্রোগ্রামিং এবং আমার পরবর্তি টিউন গুলো বুঝতে পারবেন না।

কয়েকটি প্রোগ্রামিং পরিভাষা

সি প্রোগ্রামের বেশ কিছু শব্দ ব্যবহৃত হয় যার অর্থ ও ব্যবহার প্রোগ্রামিং এ হাত দেওয়ার আগে জানা আবশ্যিক। যে কোন প্রোগ্রামিং এই শব্দগুলোর অর্থ জানা আবশ্যিক। আশা করি এগুলো পরবর্তিতে কাজে লাগবে।

Escape sequence:

C প্রোগ্রামিং ল্যাসুয়েজে কিছু Character আছে যেগুলো Out Put এ দেখা যায় না। এ সব Character কে Escape sequence বা মুক্ত ক্রম বলে। এগুলো সাধারণত একটা Backslash (\) দিয়ে আরম্ভ হয়। যেমন আমাদের যখন নতুন লাইন দরকার তখন \n দিতে হয়। আবার যখন Tab এর প্রয়োজন হয় তখন \t লিখতে হয়। ইত্যাদি।

নিচের চিত্রে সর্বাধিক ব্যবহৃত কিছু Escape sequence এর তালিকা এবং এদের কাজ দেওয়া হল।

Sequence	Purpose
\?	Question Mark
\n	New line
\r	Used to have the cursor at the beginning of the current line
\t	Brings the cursor to the next tab stop
\a	Sounds the alert noise
\\	Allows to insert backslash in a quoted expression
\'	Used to insert a single quote inside quotes
\"	Inserts double quote
\v	Vertical tab

Variables:

Variable বলতে চলক কে বুঝায়। আমরা অঙ্কে যেমন X, y অথবা Z ব্যবহার করতাম এখানে ও একই উদ্দেশ্য Variable ব্যবহৃত হয়।

যেমন: $x=3$; $y=5$; z ; ইত্যাদি। এখানে x একটি চলক বা Variable যার মান 3 ধরা হয়েছে। তেমনি y এর মান ধরা হয়েছে 5 এবং z এর কোন মান ধরা হয় নি।

Declaration:

Variable গুলো বা অন্য কোন character কোন Data type তা প্রকাশ করা হচ্ছে Declaration এর কাজ। Declaration দ্বারা কোন কিছু ঘোষণা করা হয়। অর্থাৎ কোন কিছু declare করতে Declaration ব্যবহার করা হয়।

যেমন: `int x=45; float z=2.3; char tech_tune;` ইত্যাদি।

এখানে x একটি integer type variable ঘোষণা করা হয়েছে যার মান 45 ধরা হয়েছে, z floating point variable যার মান 2.3 ধরা হয়েছে, `tech_tune` character type variable যার কোন মান ধরা হয়নি।

Expression:

Expression দ্বারা অভিব্যক্তি প্রকাশ করা হয়। এগুলো একটি Single data item প্রকাশ করে। যার মধ্যে operator sign থাকে। ($+$ $-$ $*$ $/$ $<$ $>$ $<=$ $>=$ ইত্যাদিকে operator sign বলে।)

যেমন: $x+y$; $x+y=z$; $x<y$ ইত্যাদি। এগুলো মনে হয় ব্যাখ্যা করার প্রয়োজন নেই।

খিওরি গুলো বুঝার জন্য একটি প্রোগ্রাম লিখে সব ব্যাখ্যা করে দিচ্ছি।

```
#include<stdio.h>
main()
{
    int x=3;
    int y=5;
    int z;
    z=x+y;
    printf("%d\n", z);
}
```

OUT PUT :

8

এখানে x y z তিনটা variable বা চলক।

এবং x একটি integer type variable ঘোষণা করা হয়েছে যার মান ধরা হয়েছে ৩। একই ভাবে y এর মান ধরা হয়েছে 5, এবং z এর কোন মান ধরা হয়নি।

তারপর $z=x+y$; Expression দ্বারা x+y এর মান z এর সমান ধরা হয়েছে।

আর `printf(“%d\n”, z); statement` এর ভিতরে Escape sequence(\n) দ্বারা new line প্রিন্ট করার জন্য বলা হয়েছে।

এখন এই প্রোগ্রামটা RUN করলে OUT PUT দেবে 8

9.0-ভেরিয়েবলের আউটপুট প্রকাশ পদ্ধতি: সি প্রোগ্রামিং টিউটোরিয়াল

ভেরিয়েবলের আউটপুট অপারেশন:

প্রোগ্রাম ডেটা তথা ভেরিয়েবলের মান প্রদর্শন করাকে আউটপুট অপারেশন বলে। সি প্রোগ্রামে কোন ভেরিয়েবলের মান মনিটরের পর্দায় প্রদর্শনের জন্য `printf ()`, `putc ()`, `puts ()`, `putchar ()` প্রভৃতি লাইব্রেরি ফাংশন ব্যবহার করা হয়।

printf () ফাংশন:

`printf ()` একটি লাইব্রেরি ফাংশন, যার হেডার ফাইল হল `stdio.h`, এর জন্য প্রোগ্রামের শুরুতে `#include<stdio.h>` স্টেটমেন্টের মাধ্যমে মাধ্যমে এই হেডার ফাইল সংযুক্ত করা হয়েছে। `printf ()` ফাংশনের প্রথম বন্ধনীর মধ্যে ডবল কোটেশনের (“ ”) মধ্যে যা লেখা হয় প্রোগ্রামে নিবাহে তা মনিটর স্ক্রীনে তথা আউটপুট ডিভাইসে প্রদর্শিত হয়। নিচে একটি প্রোগ্রাম দেওয়া হল:

এখানে `printf ()` ফাংশনের ডবল কোটেশনের (“ ”) মধ্যে `Welcome to C Programming.` লেখা হয়েছে। যখন প্রোগ্রামটি রান করা হয়েছে তখন আউটপুট এ সেই লেখাটি দেখিয়েছে। প্রোগ্রামটি তে কোন ভুল আছে কিনা তা দেখার জন্য `Alt + F9` চাপুন আর প্রোগ্রামটি রান করার জন্য `Ctrl + F9` চাপুন। তাহলেই প্রোগ্রামটি রান হবে।

ব্যাকস্লাশ ক্যারেক্টার:

বিশেষ কিছু ক্যারেক্টার আছে (যেমনঃ- \, \“, \n, \r, \t ইত্যাদি) যেগুলো `printf ()` ফাংশনের ডাবল কোটেশনের (“ ”) এর মধ্যে যেভাবে রাখা হয় হুবহু সেভাবে ফলাফল প্রকাশ করে না। এখানে কিছু ব্যাকস্লাশ ক্যারেক্টারের বর্ণনা দেওয়া হল:-

ব্যাকস্লাশ ক্যারেক্টার	ব্যবহার
\n	আউটপুট পরবর্তী (নতুন) লাইনের শুরুতে প্রদর্শনের জন্য।
\r	আউটপুট পূর্ববর্তী লাইনের শুরুতে একই কলাম বরাবর প্রদর্শনের জন্য।
\t	আউটপুট ডান দিকে এক ট্যাব দূরত্বে প্রদর্শনের জন্য।
\“	ডাবল কোটেশন ক্যারেক্টার প্রদর্শনের জন্য।
\a	সংকেত প্রদানের জন্য।

এখানে \n দেওয়ার জন্য নতুন একটি লাইন তৈরি হয়েছে।

ফরম্যাট স্পেসিফায়ার:- printf () ফাংশন দ্বারা কোন ডেটা বা ভেরিয়েবলের মান প্রদর্শনের জন্য নির্দিষ্ট ফরম্যাটের কতগুলো ক্যারেক্টার ব্যবহৃত হয় সেগুলোকে ফরম্যাট স্পেসিফায়ার বলা হয়। যেমন: int টাইপ ডেটা বা ভেরিয়েবলের মান প্রদর্শনের জন্য ফরম্যাট স্পেসিফায়ার হল %i বা %d, char টাইপ ডেটার জন্য %c ইত্যাদি।

ফরম্যাট স্পেসিফায়ার	ব্যবহার
%c	Char টাইপ মান ইনপুট/আউটপুট করার জন্য।
%d	Integer টাইপ মান ইনপুট/আউটপুট করার জন্য।
%f	Float টাইপ মান ইনপুট/আউটপুট করার জন্য।
%lf	Double টাইপ মান ইনপুট/আউটপুট করার জন্য।
%o	Octal টাইপ মান ইনপুট/আউটপুট করার জন্য।

এই প্রোগ্রামটি দ্বারা a, b, ও c যোগ করা হয়েছে।

সি প্রোগ্রামে পূর্ণসংখ্যা (যেমন: ২৩, ৫৬০০, -৯৮৪৫ ইত্যাদি) নিয়ে কাজ করার জন্য integer টাইপের ভেরিয়েবল ব্যবহার করা হয়। int টাইপ ভেরিয়েবল ঘোষণার জন্য int কীওয়ার্ড ব্যবহৃত হয়।

এখানে,

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

```
#include<conio.h>
```

উপরের দুই লাইন হলো হেডার ফাইল।

```
main()
```

```
{
```

দ্বিতীয় বন্ধনী দেওয়ার পর প্রোগ্রামের বডি লেখা হয়।

```
Int sum;
```

Int দিয়ে বোঝানো হচ্ছে যে আমার যে ডাটা নিয়ে কাজ করব তা পূর্ণ সংখ্যা হবে বা int এবং sum দ্বারা ভেরিয়েবল বোঝানো হয়েছে যার মধ্যে যোগফলটি থাকবে।

```
Int a,b,c;
```

a, b, ও c এর মান int টাইপের হবে তা বোঝানো হয়েছে। এবং পরে এদের মান দেওয়া হয়েছে। আপনি চাইলে এক সাথেই ভেরিয়েবলের মান ঘোষণা করতে পারেন। আর প্রতি প্রোগ্রামের শেষে অবশ্যই সেমিকলন (;) দিতে হবে। না হলে প্রোগ্রামে ভুল ধরবে।

এখানে a, b ও c ভেরিয়েবলের পরে আলাদা ভাবে

```
a=32;
```

```
b=23;
```

```
c=33; মান নেওয়া হয়েছে। আপনি চাইলে একসাথে মান নিতে পারেন বা int a=32,b=23,c=33; ।
```

```
sum=a+b+c;
```

এই লাইন দ্বারা a, b, ও c কে যোগ করে sum এর মধ্যে থাকবে তা দেখানো হয়েছে।

```
printf("%d", sum);
```

এই লাইনের জন্য আউটপুটে বা মনিটরে a, b, ও c এর যোগ ফল দেখাবে। %d দেওয়া হয়েছে ভেরিয়েবল গুলো হল int টাইপের সেই জন্য।

```
getch();
```

এই ফাংশন দ্বারা আউটপুটে মান অনেকক্ষণ ধরে রাখার জন্য দেওয়া হয়। যদি এই ফাংশন দেওয়া না হত তাহলে ও আউটপুটে রেজাল্ট দেখাত। তবে এত কম সময় যে বোঝা যেত না। তাই এই ফাংশন ব্যবহার করা হয়।

```
}
```

আর প্রোগ্রাম শেষ করতে দ্বিতীয় বন্ধনী ব্যবহার করা হয়।

9.1- পিএইচপি অনুশীলন ৬ – ভেরিয়েবল যুক্ত করার বিশেষ পদ্ধতি

1. ডাইনামিক এবং ইন্ট্রাকটিভ ওয়েব সাইট তৈরির জন্য PHP একটা শক্তিশালি ল্যঙ্গুয়েজ। PHP কে বলা হয় Hypertext Preprocessor। ওপেন সোর্স জেনারেল পারপোজ স্ক্রিপ্টিং ল্যঙ্গুয়েজ হিসেবে PHP ব্যাপকভাবে ব্যবহৃত হচ্ছে, বিশেষ করে ওয়েব ডেভলপমেন্টে এর জুরি নেই। ডাইনামিক ওয়েব সাইট তৈরির ক্ষেত্রে PHP আজ জনপ্রিয়তার শীর্ষে উঠে এসেছে। এছাড়া PHP ওয়েবে তথ্য সংরক্ষণের ক্ষেত্রে সর্বোচ্চ নিরাপত্তা প্রদান করে।হয়।এছাড়াও PHP এর সাথে MySql ডাটাবেজ সংযুক্ত করে যে কোন ওয়েব সাইটকে অনেক বেশি ব্যবহার বান্ধব করা সম্ভব হয়।তাই একজন ভাল মানের ওয়েব ডেভলপার হিসেবে নিজেকে তৈরি করার জন্য PHP শেখার বিকল্প নেই।
2. আমরা প্রজেক্ট ভিত্তিক টিউটোরিয়ালের মাধ্যমে PHP এর গুরুত্বপূর্ণ বিষয় গুলোকে আকর্ষণীয় ভাবে উপস্থাপন করার চেষ্টা করব, যেন যে কেউ খুব সহজেই PHP তে দক্ষতা অর্জন করতে পারেন।সকলের সার্বিক সহযোগিতা কমণা করছি।
3.



4. ধারাবাহিক পিএইচপি টিউটোরিয়ালের ৬ষ্ঠ পর্ব [পি এইচ পি অনুশীলন ৫](#) – [ভেরিয়েবল যুক্ত করার পদ্ধতি](#) তে আমরা পিএইচ পি তে ভেরিয়েবল যুক্ত করার পদ্ধতি দেখেছি। আজ আমরা পিএইচ পি তে ভেরিয়েবল যুক্ত করার ক্ষেত্রে যে বিষয় সমূহ বিবেচনা করতে হয় তা নিয়ে আলোচনা করব।
5. পি এইচ পি তে বিভিন্ন উপায়ে ভেরিয়েবল ঘোষণা করা যায়। কিছু বাড়তি সুবিধা পাবার জন্য ভেরিয়েবল ঘোষণা করার ক্ষেত্রে বেশ কিছু বিষয় বিবেচনায় আনতে হয়। কিছু ভেরিয়েবলের উদাহরণ বিশ্লেষণ করা যাক।
6. \$var, \$Var, \$helloWorld, \$hello_world, \$hello-world, \$helloworld, \$_helloworld, \$__helloworld যদিও উদাহরণের সবগুলো ভেরিয়েবলই পি এইচ পি তে ব্যবহার করতে কোন বাধা নেই, কিন্তু \$hello-world এবং \$__helloworld আমাদেরকে বিভ্রান্ত করতে পারে। \$hello-world এতে ড্যাস চিহ্ন না বিয়োগ চিহ্ন ব্যবহার করা হয়েছে তা একবার দেখে বলা কঠিন। \$_helloworld এর ক্ষেত্রে দুটি আন্ডারস্কোর ব্যবহার করা হয়েছে ইহাও একবার দেখে বোঝা কঠিন। তাই \$var, \$hello_world, \$helloworld, \$_helloworld এর অনুরূপ ভেরিয়েবল ব্যবহার করা উচিত।

7. আজকের প্রজেক্ট

```
<html>
<head>
<title>www.tutorialbd.com </title>
</head>
<body>
<?php
$num=25;
echo $num;
echo "<br />";
$web_site="<h2>www.tutohost.com</h2>";
echo $web_site;
echo "<br />";
$web_Site="We are relaibale Bangladeshi hosting provider. The world wide
technical and support team is working for your best movement. We are
dedicated with client requirement.";
echo $web_Site;
echo "<br />";
echo "<br />";
$num=459;
```



```

echo $num;
?>
</body>
</html>

```

18. উপরের কোডটুকু একটা নোটপ্যাডে লেখার পর নোটপ্যাডের File মেনুতে ক্লিক করে Save as এ ক্লিক করার পর File name এ Variable.php দিয়ে Save as type এর ড্রপ ডাউন হতে All files সিলেক্ট করার পর Save এ ক্লিক করে Save করতে হবে।
19. index.php ফাইলটিকে ব্রাউজারের মাধ্যমে প্রদর্শন করার পূর্বে Variable.php ফাইলটিকে C:\xampp\htdocs\ অর্থাৎ\htdocs ফোল্ডারের মধ্যে রাখতে হবে। ব্রাউজারের Variable.php মাধ্যমে প্রদর্শন করার পূর্বে আমাদের নিশ্চিত হয়ে নিতে হবে xampp অর্থাৎ Apache সার্ভার Running অবস্থায় আছে।
20. xampp Running অবস্থায় থাকলে first.php ফাইলটি ব্রাউজারে দেখানোর জন্য এড্রেস বারে লেখতে হবে <http://localhost/variable.php> ।
21. তাহলে ব্রাউজারে নিচের ছবির মত দেখাবে।

22.

23. প্রজেক্ট সম্পর্কে কিছু কথা

24. \$num=25; প্রকাশ করে num নামে একটা ভেরিয়েবল ঘোষণা করা হয়েছে।
25. echo \$num; প্রকাশ করে num নামে তৈরি ভেরিয়েবল এর মান 25 ব্রাউজারে প্রদর্শন করবে।
26. echo "
"; প্রকাশ করে ব্রাউজারে প্রদর্শনের সময় একটা লাইন ব্রেক হবে।
27. \$web_site="<h2>www.tutohost.com</h2>"; এখানে \$web_site ভেরিয়েবলটিতে নামের S টি lower-case ব্যবহার করা হয়েছে ।
28. echo \$web_site; প্রকাশ করে web_site নামে তৈরি ভেরিয়েবল এর মান হিসেবে প্রদত্ত স্ট্রিং www.tutohost.com লেখাটি ব্রাউজারে প্রদর্শন করবে।
29. \$web_Site="We are reliable Bangladeshi hosting provider. The world wide technical and support team is working for your best movement. We are dedicated with client requirement."; এখানে \$web_Site ভেরিয়েবলটিতে নামের S টি Upper-case ব্যবহার করা হয়েছে।
30. echo \$web_Site; প্রকাশ করে \$web_Site নামে তৈরি ভেরিয়েবল এর মান হিসেবে প্রদত্ত স্ট্রিং We are reliable Bangladeshi hosting provider. The world wide technical and support team is working for your best movement. We are dedicated with client requirement. লেখাটি ব্রাউজারে প্রদর্শন করবে। আমরা এতক্ষণে বুঝে গিয়েছি যে PHP তে ভেরিয়েবল এর case-sensitive হয়।
31. \$num=459; সর্ব প্রথমে আমরা num নামে একটা ভেরিয়েবল ঘোষণা করে মান দিয়েছিলাম 25 , এখন আমরা এটার মান দিয়ে দিচ্ছি 459 ।

echo \$num; প্রকাশ করে num নামে তৈরি ভেরিয়েবলের এর নতুন মান 459 ব্রাউজারে প্রদর্শন করবে। অর্থাৎ একই নাম ব্যবহার করে বিভিন্ন পর্যায়ে একাধিক ভেরিয়েবল ঘোষণা করলে ভেরিয়েবলের সর্বশেষ মানটি ব্রাউজারে প্রদর্শিত হবে।

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

```
main ()
```

```
{
```

```
P r i n t f ("THE END");
```

```
}
```

O u t p u t :

T H E E N D