58. Date and Time Function In C Programming:

Time and date function is used to use the current date and time as per the user requirement. If a programmer want to show the current date and time to his application then he can use the date and time function. In C programming there are two header files are there which are used for date and time operation. First one is the “#include<time.h>” and second one is “#include<windows.h>”.

TIMEDATE.c

```c
#include<stdio.h>
#include<conio.h>
#include<time.h>

int main()
{
    time_t t = time(NULL);
    struct tm date = *localtime(&t);

    date.tm_mday;
    date.tm_mon;
    date.tm_year;

    date.tm_hour;
    date.tm_min;
    date.tm_sec;

    printf("current date : %d-%d-%d\n",date.tm_year+1900,date.tm_mon+1,date.tm_mday);
    printf("Current Time : %d:%d:%d\n",24-date.tm_hour,date.tm_min,date.tm_sec);
}
```

```
C:\Users\Atish kumar sahu\De
current date : 2023-2-5
Current Time : 2:26:11

--------------------------------
Process exited after 0.05982 seconds with return value 0
Press any key to continue . . .
```

TIMEDATE.c

```c
#include<stdio.h>
#include<conio.h>
#include<windows.h>
int main()
{
    SYSTEMTIME stime;
    GetSystemTime(&stime);
    GetLocalTime(&stime);

    printf("Current date : %d/%d/%d\n",stime.wYear,stime.wMonth,stime.wDay);
    printf("Current time : %d:%d:%d\n",stime.wHour,stime.wMinute,stime.wSecond);
}
```

```
C:\Users\Atish kumar sahu\De
Current date : 2023/2/5
Current time : 22:28:25

--------------------------------
Process exited after 0.06755 seconds with return value 0
Press any key to continue . . .
```

TIMEDATE.c

```c
#include<stdio.h>
#include<conio.h>
#include<time.h>
#include<windows.h>

int main()
{
    time_t t = time(NULL);
    struct tm date = *localtime(&t);

    date.tm_mday;
    date.tm_mon;
    date.tm_year;

    date.tm_hour;
    date.tm_min;
    date.tm_sec;

    printf("current date : %d-%d-%d\n",date.tm_year+1900,date.tm_mon+1,date.tm_mday);
    printf("Current Time : %d:%d:%d\n",24-date.tm_hour,date.tm_min,date.tm_sec);

    SYSTEMTIME stime;
    GetSystemTime(&stime);
    GetLocalTime(&stime);

    printf("Current date : %d/%d/%d\n",stime.wYear,stime.wMonth,stime.wDay);
    printf("Current time : %d:%d:%d\n",24-stime.wHour,stime.wMinute,stime.wSecond);
}
```

```
C:\Users\Atish kumar sahu\De
current date : 2023-2-5
Current Time : 2:32:44
Current date : 2023/2/5
Current time : 2:32:44

--------------------------------
Process exited after 0.07024 seconds with return value 0
Press any key to continue . . .
```

59. Command Line Arguments In C Programming:

The arguments passed from command line are called command line arguments. These arguments are handled by main() function.

it is an important concept in c programming. sometimes we need to pass arguments from the command line to the program a set of inputs. command line arguments are used to supply parameters to the program when it is invoked. it is mostly used when you need to control your program from this console. these arguments are passed to the main() method.

FFmpeg is a free and open source project consisting of a vast software suit of libraries and programs for handling video, audio and other multimedia files and streams. FFmpeg.exe is a command line utility written in C language. other examples like, git, brew, apt-get, etc.

There are various kinds of ways are there to run a program using IDE, By double click on the .exe file, using command file. you can pass arguments to the main function only when you are calling your program using command line.

The main() function is a Take Something nature of function. It can take Actual and formal arguments, main() may take arguments. Who calls main? During the execution of program the operating system will call the main() function for execution of the operations.

CMDARG.c

```c
#include<stdio.h>
#include<conio.h>
int main(int argc)
{
    printf("Argc Number Is : %d",argc);
}
```

Command Prompt

```
C:\Users\Atish kumar sahu\desktop>gcc CMDARG.c -o CMDARG.exe

C:\Users\Atish kumar sahu\desktop>CMDARG.exe
Argc Number Is : 1
C:\Users\Atish kumar sahu\desktop>CMDARG.exe 1 2 3 Atish Kumar Sahu
Argc Number Is : 7
C:\Users\Atish kumar sahu\desktop>
```

CMDARG.c

```c
#include<stdio.h>
#include<conio.h>
int main(int argc, char const *argv[])
{
    printf("Argc Number Is : %d\n",argc);
    printf("Argc Value Is : %s\n",argv[0]);
    printf("Argc Value Is : %s\n",argv[1]);
    printf("Argc Value Is : %s\n",argv[2]);
}
```

Command Prompt

```
C:\Users\Atish kumar sahu\desktop>gcc CMDARG.c -o CMDARG.exe

C:\Users\Atish kumar sahu\desktop>CMDARG.exe 1 Atish
Argc Number Is : 3
Argc Value Is : CMDARG.exe
Argc Value Is : 1
Argc Value Is : Atish

C:\Users\Atish kumar sahu\desktop>
```

CMDARG.c

```c
#include<stdio.h>
#include<conio.h>
int main(int argc, char const *argv[])
{
    for(int i = 0; i < argc; i++)
    {
        printf("Value : %s\n",argv[i]);
    }
    getch();
}
```

Command Prompt

```
C:\Users\Atish kumar sahu\desktop>gcc CMDARG.c -o CMDARG.exe

C:\Users\Atish kumar sahu\desktop>CMDARG.exe 1 2 3 A B C a b c Atish
Value : CMDARG.exe
Value : 1
Value : 2
Value : 3
Value : A
Value : B
Value : C
Value : a
Value : b
Value : c
Value : Atish

C:\Users\Atish kumar sahu\desktop>
```

CMDARG.c

```c
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
int main(int argc, char const *argv[])
{
    int num1 = atoi(argv[1]);
    int num2 = atoi(argv[2]);

    printf("Sum : %d",(num1 + num2));
    getch();
}
```

Command Prompt - CMDARG

```
C:\Users\Atish kumar sahu\desktop>gcc CMDARG.c -o CMDARG.exe

C:\Users\Atish kumar sahu\desktop>CMDARG.exe 10 20
Sum : 30
```

CMDARG.c

```c
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
int main(int argc, char const *argv[])
{
    int count = 0;

    for(int i = 1; i < argc; i++)
    {
        int num = atoi(argv[i]);
        count += num;
    }
    printf("Sum : %d",count);
    getch();
}
//atoi stands for ARGUMENT TO INTEGER.
```

Command Prompt

```
C:\Users\Atish kumar sahu\desktop>gcc CMDARG.c -o CMDARG.exe

C:\Users\Atish kumar sahu\desktop>CMDARG.exe 1 2 3 4 5 6 7 8 9 10
Sum : 55
C:\Users\Atish kumar sahu\desktop>
```