

**Shri G. S. Institute of  
Technology and Science**

CO24997 : programming practices

November 21, 2022

## **MINI PROJECT : UNIT CONVERTER :**

starting date : 13/11/22

ending date : 21/11/22

**The project is coded in two languages c and java**

### **0.1 UNIT CONVERTER :**

project is made on unit converter. Unit converter enables conversion and switching the base unit to converting unit. In this project we use 5 quantities for converting into unit, for ex. meter into kilometer and million to crores.

five quantities are used in this project  
:

1. mass : kilogram and gram
2. temperature : kelvin and celcius
3. time : seconds and hours
4. currency : million and crores
5. length : kilometer and meter

## 0.2 IN C language :

no. of code line is : **205** In unit converter we use 10 user define function are:

1. float getgram(float kilogram);
2. float getkilogram(float gram);
3. float getmillion(float crores);
4. float getcrores(float million);
5. float getkelvin(float celcius);
6. float getcelcius(float kelvin);
7. float gethours(float seconds);
8. float getseconds(float hours);
9. float getkilometer(float meter);
10. float getmeter(float kilometer);

### 0.2.1 debugging :

The screenshots of debugging of unit converter program are :

```
deepikamandroniyahp@LAPTOP-I3GJHV0P: ~
93 updates can be applied immediately.
51 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

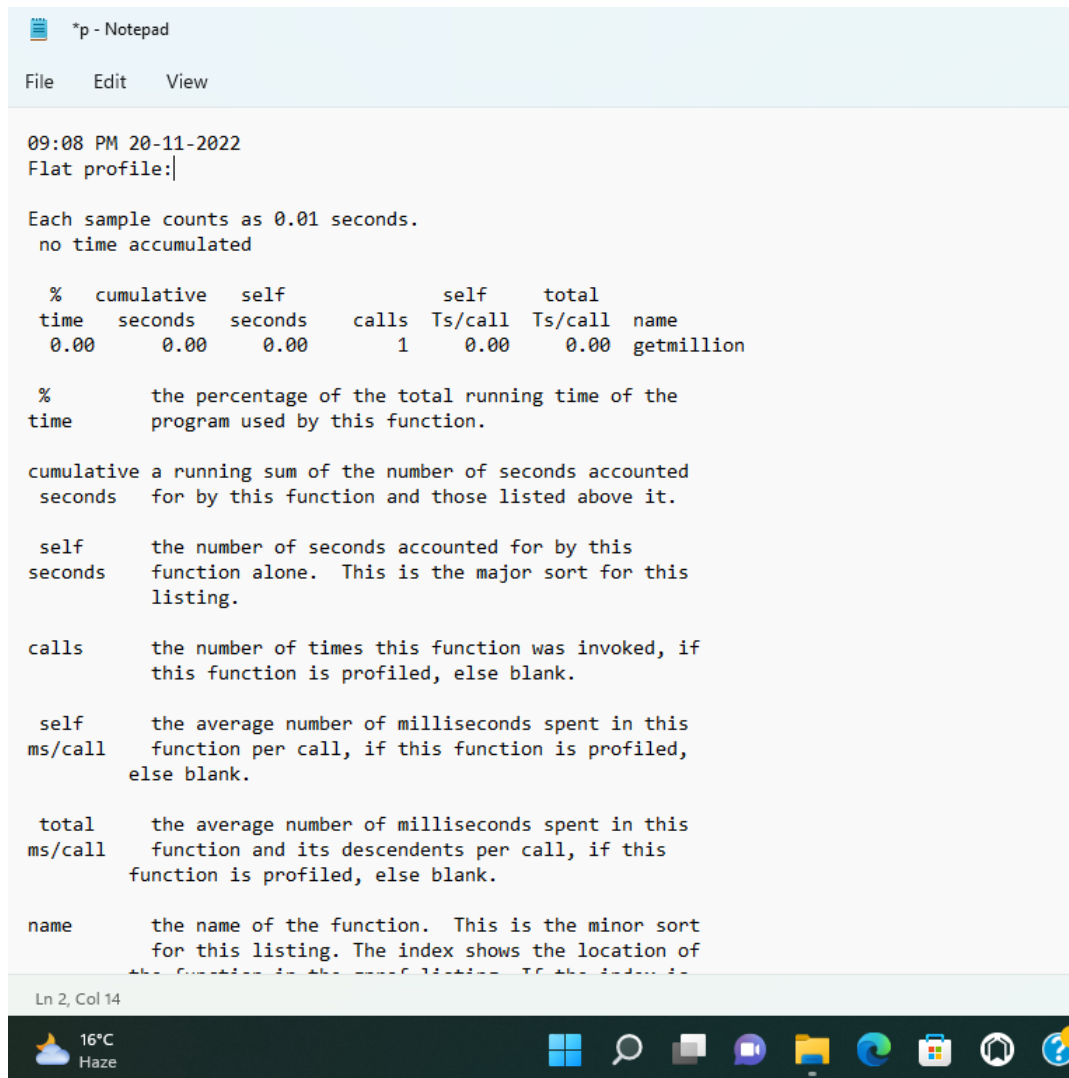
The list of available updates is more than a week old.
To check for new updates run: sudo apt update

This message is shown once a day. To disable it please create the
/home/deepikamandroniyahp/.hushlogin file.
deepikamandroniyahp@LAPTOP-I3GJHV0P:~$ project.c
project.c: command not found
deepikamandroniyahp@LAPTOP-I3GJHV0P:~$ gcc project.c
deepikamandroniyahp@LAPTOP-I3GJHV0P:~$ nano project.c
deepikamandroniyahp@LAPTOP-I3GJHV0P:~$ ./a.out
welcome to unit conversion!!
    please select the quantity you want to convert

Mass(m) , Currency(C) , Temperature(T) ,Time(t) , Length(L)
m
welcome to unit conversion!!
please choose :
    1 for getkilogram
    2 for getgram
1
enter the value in gram to get kilogram
12
the conversion of gram in kilogram is 0.012
deepikamandroniyahp@LAPTOP-I3GJHV0P:~$ █
```

## 0.2.2 profiling :

The screenshots of profiling of unit converter program are :



```
*p - Notepad
File Edit View

09:08 PM 20-11-2022
Flat profile:

Each sample counts as 0.01 seconds.
no time accumulated

% cumulative self self total
time seconds seconds calls Ts/call Ts/call name
0.00 0.00 0.00 1 0.00 0.00 getmillion

% the percentage of the total running time of the
time program used by this function.

cumulative a running sum of the number of seconds accounted
seconds for by this function and those listed above it.

self the number of seconds accounted for by this
seconds function alone. This is the major sort for this
listing.

calls the number of times this function was invoked, if
this function is profiled, else blank.

self the average number of milliseconds spent in this
ms/call function per call, if this function is profiled,
else blank.

total the average number of milliseconds spent in this
ms/call function and its descendents per call, if this
function is profiled, else blank.

name the name of the function. This is the minor sort
for this listing. The index shows the location of
the function in the source file. To the left of the name is
the line number of the function in the source file.

Ln 2, Col 14
```

```
*p - Notepad
File Edit View

function is profiled, else blank.

name      the name of the function. This is the minor sort
           for this listing. The index shows the location of
           the function in the gprof listing. If the index is
           in parenthesis it shows where it would appear in
           the gprof listing if it were to be printed.

Copyright (C) 2012-2022 Free Software Foundation, Inc.

Copying and distribution of this file, with or without modification,
are permitted in any medium without royalty provided the copyright
notice and this notice are preserved.

Call graph (explanation follows)

granularity: each sample hit covers 4 byte(s) no time propagated

index % time    self  children   called    name
              0.00   0.00      1/1      main [16]
[1]      0.0    0.00   0.00      1      getmillion [1]
-----

This table describes the call tree of the program, and was sorted by
the total amount of time spent in each function and its children.

Each entry in this table consists of several lines. The line with the
index number at the left hand margin lists the current function.
The lines above it list the functions that called this function,
and the lines below it list the functions this one called.
This line lists:

Ln 2, Col 14
16°C
Haze
```

\*p - Notepad

File Edit View

-----

This table describes the call tree of the program, and was sorted by the total amount of time spent in each function and its children.

Each entry in this table consists of several lines. The line with the index number at the left hand margin lists the current function. The lines above it list the functions that called this function, and the lines below it list the functions this one called. This line lists:

index A unique number given to each element of the table.  
Index numbers are sorted numerically.  
The index number is printed next to every function name so it is easier to look up where the function is in the table.

% time This is the percentage of the `total' time that was spent in this function and its children. Note that due to different viewpoints, functions excluded by options, etc, these numbers will NOT add up to 100%.

self This is the total amount of time spent in this function.

children This is the total amount of time propagated into this function by its children.

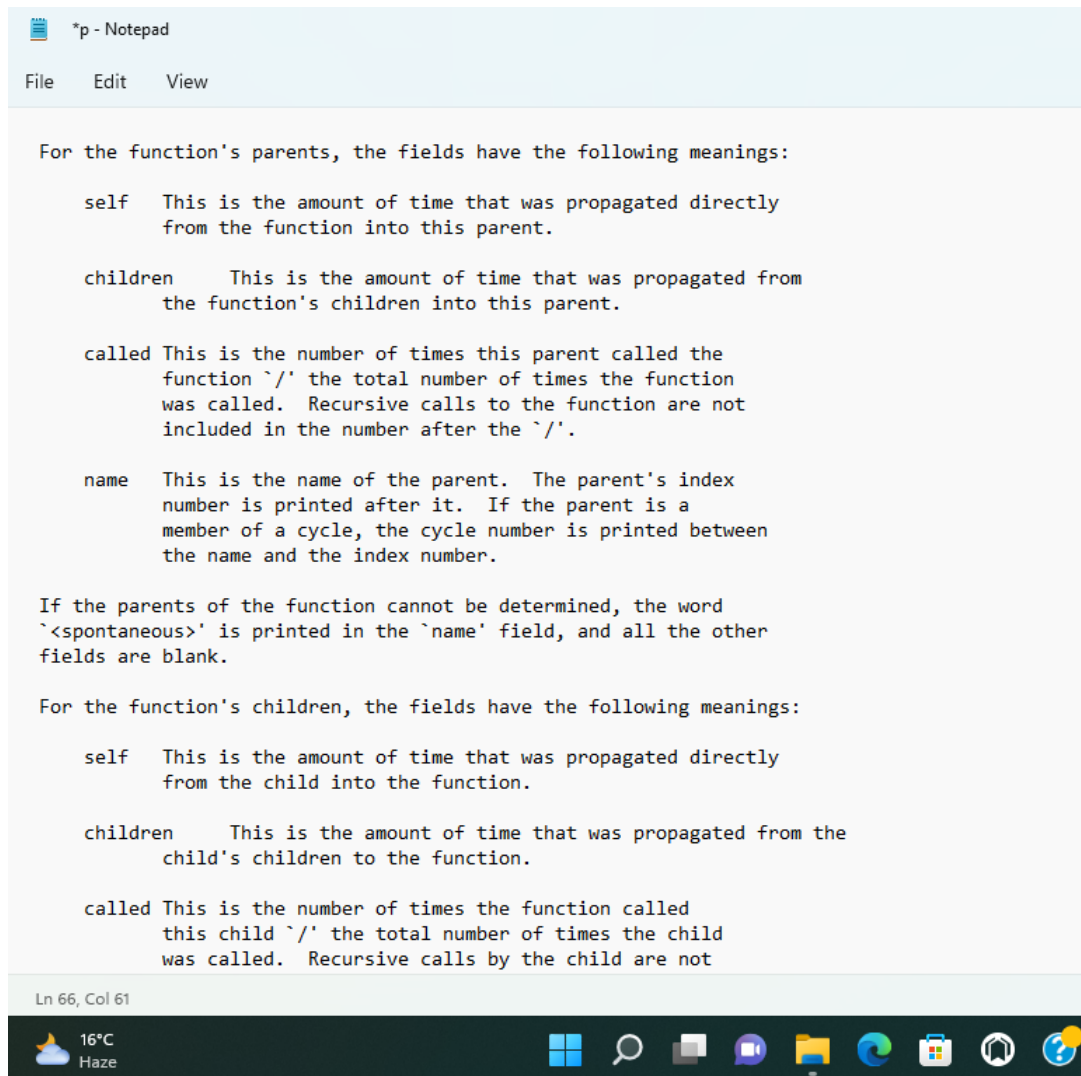
called This is the number of times the function was called. If the function called itself recursively, the number only includes non-recursive calls, and is followed by a `+' and the number of recursive calls.

name The name of the current function. The index number is printed after it. If the function is a member of a cycle, the cycle number is printed between the function's name and the index number.

Ln 66, Col 61

16°C Haze

Windows taskbar icons: Start menu, Search, Task View, Microsoft Edge, File Explorer, Calendar, Photos, Settings, Help and Support.



```
*p - Notepad
File Edit View

For the function's parents, the fields have the following meanings:

self This is the amount of time that was propagated directly
      from the function into this parent.

children This is the amount of time that was propagated from
          the function's children into this parent.

called This is the number of times this parent called the
        function '/' the total number of times the function
        was called. Recursive calls to the function are not
        included in the number after the '/'.

name This is the name of the parent. The parent's index
      number is printed after it. If the parent is a
      member of a cycle, the cycle number is printed between
      the name and the index number.

If the parents of the function cannot be determined, the word
'<spontaneous>' is printed in the 'name' field, and all the other
fields are blank.

For the function's children, the fields have the following meanings:


self This is the amount of time that was propagated directly
      from the child into the function.

children This is the amount of time that was propagated from the
          child's children to the function.

called This is the number of times the function called
        this child '/' the total number of times the child
        was called. Recursive calls by the child are not

Ln 66, Col 61
```

16°C Haze





```
*p - Notepad
File Edit View

from the child into the function.

children This is the amount of time that was propagated from the
child's children to the function.

called This is the number of times the function called
this child '/' the total number of times the child
was called. Recursive calls by the child are not
listed in the number after the '/'.

name This is the name of the child. The child's index
number is printed after it. If the child is a
member of a cycle, the cycle number is printed
between the name and the index number.

If there are any cycles (circles) in the call graph, there is an
entry for the cycle-as-a-whole. This entry shows who called the
cycle (as parents) and the members of the cycle (as children.)
The '+' recursive calls entry shows the number of function calls that
were internal to the cycle, and the calls entry for each member shows,
for that member, how many times it was called from other members of
the cycle.

Copyright (C) 2012-2022 Free Software Foundation, Inc.

Copying and distribution of this file, with or without modification,
are permitted in any medium without royalty provided the copyright
notice and this notice are preserved.

Index by function name

[1] getmillion

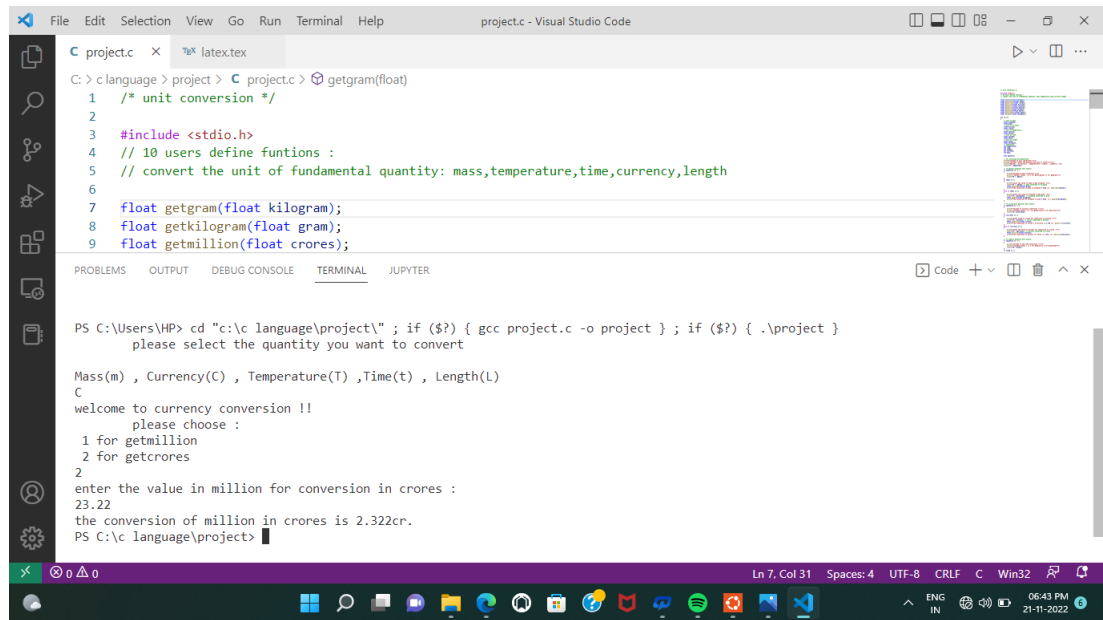
Ln 66, Col 61
```

16°C Haze

Windows taskbar icons: Start, Search, Task View, Microsoft Edge, File Explorer, Calendar, Mail, Photos, Settings, Help and Support, and a red notification icon.

### 0.2.3 CODE OUTPUT :

The screenshoot of c code output are :



The screenshot displays the Visual Studio Code interface with a C program named 'project.c' open in the editor. The code defines three functions: 'getgram' for converting kilograms to grams, 'getkilogram' for converting grams to kilograms, and 'getmillion' for converting millions to crores. The terminal window shows the execution of the program, which prompts the user to select a quantity for conversion. The user chooses '1' for 'getmillion', enters '23.22' as the value in millions, and the program outputs 'the conversion of million in crores is 2.322cr.'.

```
C: > c language > project > C project.c > getgram(float)
1  /* unit conversion */
2
3  #include <stdio.h>
4  // 10 users define funtions :
5  // convert the unit of fundamental quantity: mass,temperature,time,currency,length
6
7  float getgram(float kilogram);
8  float getkilogram(float gram);
9  float getmillion(float crores);
```

```
PS C:\Users\HP> cd "c:\c language\project\" ; if ($?) { gcc project.c -o project } ; if ($?) { .\project }
please select the quantity you want to convert

Mass(m) , Currency(C) , Temperature(T) ,Time(t) , Length(L)
C
welcome to currency conversion !!
please choose :
1 for getmillion
2 for getcrores
2
enter the value in million for conversion in crores :
23.22
the conversion of million in crores is 2.322cr.
PS C:\c language\project>
```

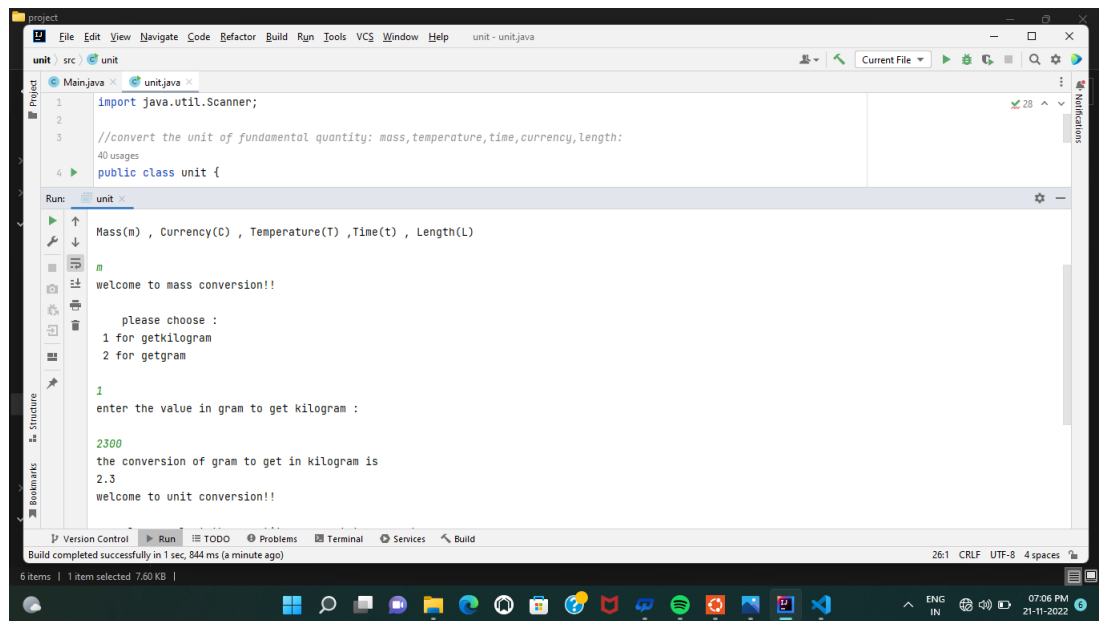
## 0.3 In java language :

no. of code line is : **168**

The project made with 10 user define function in c language and now we convert into java language:

### 0.3.1 CODE OUTPUT :

The screenshoot of java code output are :



The screenshot displays an IDE window with a Java file named 'unit.java'. The code defines a 'unit' class with methods for mass, currency, temperature, time, and length conversions. The 'Run' console shows the program's execution, including a welcome message, a choice of conversion type, and a specific mass conversion example from grams to kilograms.

```
import java.util.Scanner;

//convert the unit of fundamental quantity: mass,temperature,time,currency,length:
40 usages
public class unit {

    Mass(m) , Currency(C) , Temperature(T) ,Time(t) , Length(L)

    welcome to mass conversion!!

    please choose :
    1 for getkilogram
    2 for getgram

    1
    enter the value in gram to get kilogram :

    2300
    the conversion of gram to get in kilogram is
    2.3
    welcome to unit conversion!!
```

**END OF REPORT**