FACULTY OF ENGINEERING AND TECHNOLOGY
UNIVERSITY OF LUCKNOW
LUCKNOW
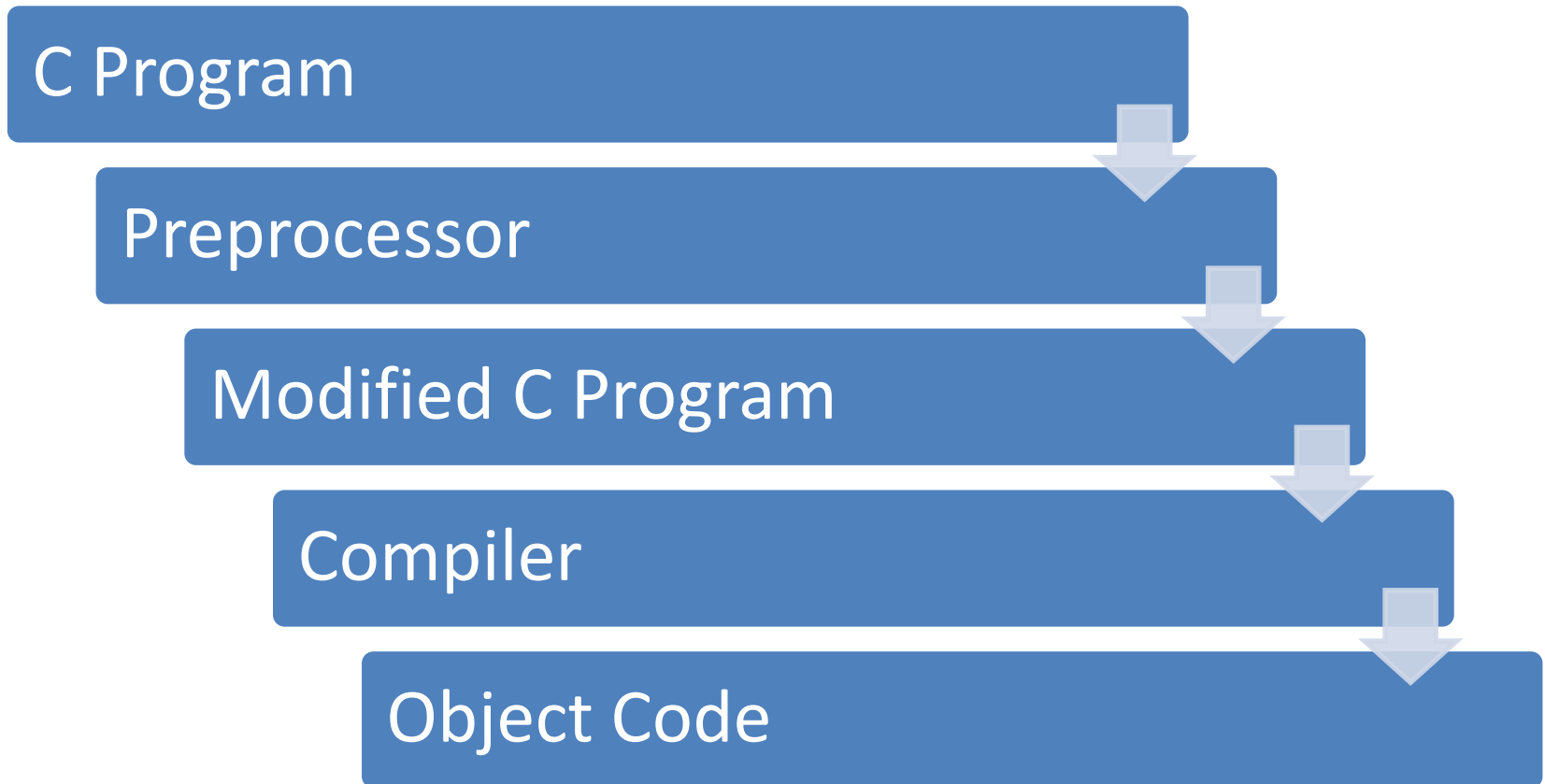

Computer System and Programming in 'C'

CS-101/201

Er. Zeeshan Ali Siddiqui

Assistant Professor

Deptt. of C.S.E.

PREPROCESSOR

Overview

- The C preprocessor is a macro processor that is used automatically by the C compiler to *transform* our program before actual compilation. It is called a macro processor because it allows you to define macros, which are brief abbreviations for longer constructs.

C Program

Preprocessor

Modified C Program

Compiler

Object Code

Types

- Three kinds of directives

- File inclusion
  - #include

- Macros
  - #define

- Conditional compilation
  - #if, #ifdef, #ifndef, #elif, #else, #endif

Rules

- Always starts with a line with "#"

- It can appear *anywhere* in a program

- Takes *one* line unless explicitly continue

  - #define MAX_SIZE 100

  - #define MAX_FILES \
    100

FILE INCLUSION

File Inclusion

- It allow a program or a module's implementation to use certain *Header* files

- A header file contains *declarations* for a module

    -Name of the header file should end in .h

- User-define header files are placed in **" ..."**

    ➤ For example:

    ➤ #include "myfile.h"

- System header files are placed in: < ... >

    ➤ For example:

    ➤ #include <stdio.h>

MACROS

Types

- Two kinds of macros

  - Object-like macro

  - Function-like macro

Object-like Macro

- It provide the facility of parameterized text substitution.

- Syntax:
  - #define identifier string

- Macro definition:
  - #define max_size 100

- Macro replacement:
  - int arr[max_size];

    Becomes
  - int arr[100];

Sample Program-1

```c
/*Program to take array elements from the user and print those elements*/
#include<stdio.h>
#define max_size 100
int main()
{
    int marks[max_size],n,i;
    printf("Please enter number of elements (1<=n<=100)\n");
    scanf("%d",&n);
    if(n<=0||n>100)
    {
        printf("\nWrong input provided!\n Please enter the value in given range\n\n");
        main();
    }
    else
    {
        //Taking values from the user start
        printf("Please enter %d array elements\n",n);
        for(i=0;i<n;i++)
            scanf("%d",&marks[i]);
        //Taking values from the user end
        //Printing values from the user start
        printf("You have entered\n");
        for(i=0;i<n;i++)
            printf("%d\n",marks[i]);
        //Printing values from the user end
    }
    return 0;
}
```

```
D:\DEVC\macro1.exe

Please enter number of elements (1<=n<=100)
3
Please enter 3 array elements
25
36
45
You have entered
25
36
45
------------------------------------
Process exited after 3.713 seconds with return value 0
Press any key to continue . . .
```

Function-like Macro

- **Syntax:**

#define identifier (arg1, arg2, ... argn) string

- **Example 1:**

```
#define addone(x) x+1

i = 2*addone(2);
becomes
i = 2*2+1
```

← output: 5

- **Example 2:**

```
#define addone(x) ((x)+1)
i = 2*addone(2);
becomes
i = 2*((2)+1)
```
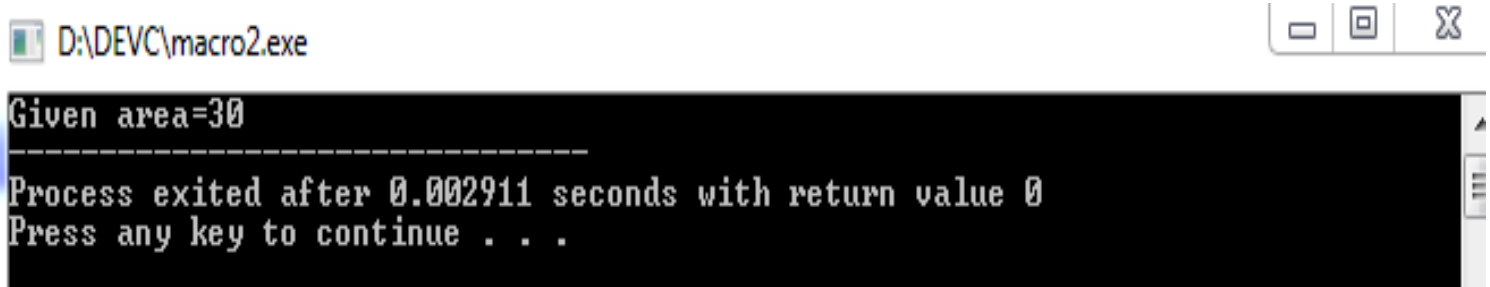
← output: 6

Sample Program-2

```c
#include<stdio.h>
#define area(x,y) (x*y)
int main()
{
printf("Given area=%d",area(5,6));
return 0;
}
```

Output

```
D:\DEVC\macro2.exe

Given area=30
---------------------------------
Process exited after 0.002911 seconds with return value 0
Press any key to continue . . .
```

Conditional Compilation

- A conditional directive is used to instruct the preprocessor to *select* whether or not to include a block of code in the final token stream passed to the compiler.

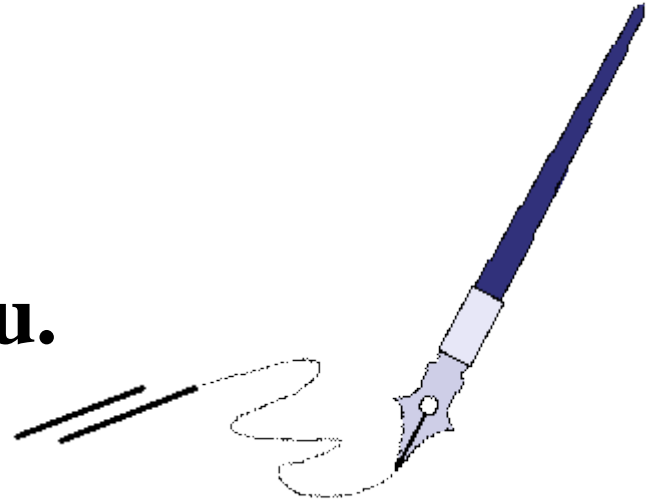- List of the useful preprocessor conditional directives

```
1  #ifdef
2  #ifndef
3  #if
4  #else
5  #elif
6  #endif
7  #error
8  #undef
```

Exercise

- What is preprocessor directive? Explain with example.

- What do you mean by macros?

- Write the process to define and call a macro.

- Explain the importance of #define preprocessor directive.

- In which condition/situation will you recommend to use conditional directives in your program.

- Give the rules for using macros.

- Give an example of function-like macro.

- Analyze the below code and give your reasoning:

```c
#include<stdio.h>
#define foet(i)\
printf("May God bless you.\n");
int main()
{
    foet(0);
    return 0;
}
```

Thank You.

BTQ

Find the next letter in the sequence

A, E, F, H, ___ ?