ECE 3331

Introduction
Integer/floating number
Characters

A First C Program
Real World Application: Computing Distances
Identifiers

A First C Program

```
/* This is a simple C program */
#include <stdio.h>
main()
{
   printf( “Welcome to our first C program! \n”);
}
```

The output is:

```
Welcome to our first C program!
```

Analysis

- /* beginning of a comment
- */ end of a comment
- #include <stdio.h> preprocessor
- main() a must, exe begins here
- {…} enclose a group of statements
- printf("…\n") copies to video display
- \n escape character: new line
- ; terminates a statement

A Simple Computation Code

Given speed=14 km/h

Inputs time in hours

Outputs distance traveled in km

Exits if input time <=0

The output looks like:

```
Enter next time: 2
Time = 2 hours
Distance = 28 kilometers

Enter next time: 0
*** End of Program ***
```

```
#include <stdio.h>                /*Note:   this line should appear if we use scanf()
                                            and printf ( ) function */
main()
{
     int distance, rate, time;
     rate = 14;
     printf( "Enter next time: "  );
     scanf( "%d", &time);        /* Note: %d denotes integer. ampersand  (& ) is
                                    required before each (non pointer) variable in scanf */
     while ( time > 0 ) {
              distance = rate * time;
              printf( "Time = %d hours\n", time);
              printf( "Distance = %d kilometers\n\n", distance);
              printf( "Enter next time: " );
              scanf( "%d", &time );
     }
     printf( "*** End of Program ***\n" );
}                                                 /* file name distance.c */
```

What are stored in the memory addresses?

distance	rate	time
?	?	?

rate = 14;

distance	rate	time
?	14	?

```
printf( "Enter next time: " );
scanf( "%d", &time );
```

Output: Enter next time: 2

distance	rate	time
?	14	2

Distance = rate * time

Output: Enter next time: 2

distance	rate	time
28	14	2

```
printf( “Time = %d hours\n”, time);
printf( “Distance = %d kilometers\n\n”, distance);
printf( “Enter next time: “ );
scanf(“%d”, &time);
```

Output:
Enter next time: 2
Time = 2 hours
Distance = 28 kilometers

Enter next time: 4

distance	rate	time
28	14	4

Distance = rate * time

Output is the same as above

distance	rate	time
56	14	4

```
printf( “Time = %d hours\n”, time);
printf( “Distance = %d kilometers\n\n”, distance);
printf( “Enter next time: “ );
scanf(“%d”, &time);
```

Output:

Enter next time: 2
Time = 2 hours
Distance = 28 kilometers

Enter next time: 4
Time = 4 hours
Distance = 56 kilometers

Enter next time: 0

distance	rate	time
56	14	0

While (time > 0)

Ouptput:

```
Enter next time: 2
Time = 2 hours
Distance = 28 kilometers

Enter next time: 4
Time = 4 hours
Distance = 56 kilometers

Enter next time: 0
*** End of Program ***
```

Identifiers

Identifier is the word for name in a high-level language. An identifier in C must satisfy the following requirements:

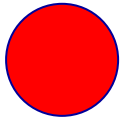
- It must start with a letter (A through Z or a through z) or the underscore character (_).

- It must consist of only letters (A through Z or a through z), digits (0 through 9), and the underscore character (_).
- It must not be a keyword. A keyword is a word such as int or while that has a special meaning. A complete list of keywords is given in page 89.

Note: uppercase characters are distinguished from lowercase characters.

Distance
_distance
total_cars
Total_cars
Total_Cars

Distance$, 2distance, float, total car, total-car



Things learned

- int a,d defines variables *a* and *d* as integers
- scanf("%d",&a)
  - read from keyboard
  - %d: decimal integer
  - &a: variable named a
  - read from the keyboard a decimal integer and store the value in the address named a
- while ( ){…}
  - while loop; repeat {…} so long as () is true

continued

- (a >0) if a>0 then the expression is true; otherwise it is false
- while {…} does not need ;
- &a in scanf to identify variable a
- a in printf