The University Of Sheffield.

Building a Well Structured Program

Using functions and managing memory

The University Of Sheffield.

Learning Outcomes

- Using and Writing Functions
- Memory management
  - Introduction to pointers
- Function calls
- Arrays and Structures
- Dynamic memory allocation
- Understanding function calls - the function pointer

The University Of Sheffield.

```
/*Use the printf function to Display a welcome message on the users screen*/
printf("Welcome to the C Language!\n);"
```

```
printf("b=%f fb=%f\n",b,fb);
```

The University Of Sheffield.

Overview the input output functions

- Printf
- Scanf
- Format specifiers
- The address operator used in scanf

The University Of Sheffield.

printf

- Provides formatted input and output
- Input for printf is a format specification followed by a list of variable names to be displayed
- Note the use of escape characters e.g. \n generates a newline

printf(“variable %d is %f\n”, myint, myfloat);

Examples

```
/*Use the printf function to Display a welcome message on the users screen*/
printf("Welcome to the C Language!\n);"
```

```
printf("b=%f fb=%f\n",b,fb);
```

The University Of Sheffield.

scanf

- Provided an input format and a list of variables
- scanf(“%d”, &myint);
- Note variable name has & in front

```
/*Request input from the user*/
printf("Enter the first integer\n");
scanf("%d", &i1);       /*Read in the integer*/
printf("Enter the second integer\n");
scanf("%d", &i2);
```

The University Of Sheffield.

Escape Characters

Escape Sequence	Description
\n	Newline, position cursor at the start of a new line
\t	Horizontal tab, move cursor to the next tab stop
\r	Carriage return. Position cursor to the beginning of the current line; do not advance to the next line.
\a	Alert, sound system warning beep
\\	Backslash, print a backslash character in a printf statement
\”	Double quote print a double quote character in a printf statement.

The University Of Sheffield.

Format Specifiers for printf and scanf

Data Type	Printf specifier	Scanf specifier
long double	%Lf	%Lf
double	%f	%lf
float	%f	%f
unsigned long int	%lu	%lu
long int	%ld	%ld
unsigned int	%u	%u
int	%d	%d
short	%hd	%hd
char	%c	%c

```c
/*Bisection method for finding roots*/

/* here is an example use of the while statement
// which is used for finding the root of a polynomial
// which is known to lie within a certain interval.
// a is the lower value of the range
// b is the upper value of the range */
#include <stdio.h>
#include <math.h>
#include <float.h>

/*
  Note FLT_MIN, FLT_MAX and FLT_EPSILON
  defined in float.h
*/

float sign(float f){return(fabs(f)/f);}

int main(char **argv, int argc)
{
/*Main routines here see next slide*/
        return 0;
}
```

The University Of Sheffield.

Practice Session

```
int main(char **argv, int argc)
{

        float x,fx;
        float a = 0;
        float fa = -FLT_MIN;
        float b = 3;
        float fb = FLT_MAX;
```

```
  while( fabs(b-a)>(FLT_EPSILON*b))
  {
    x = (a+b)/2;
    /*The function whose root is to be determined*/
    fx = pow(x,3)-2*x-5;
    if(sign(fx)==sign(fa))
    {
      a = x;
      fa = fx;
printf("a=%f fa=%f\n",a,fa);
    }
    else
    {
      b = x;
      fb = fx;
printf("b=%f fb=%f\n",b,fb);
    }

  }
  printf(" The root is :%f\n",x);
```

The University Of Sheffield.

Further Sessions

- Building Applications using Make
- From C to C++
- Boost your programming using the standard template libraries