POINTER VARIABLES

VOID POINTERS

- A void pointer is a pointer that has no associated data type with it.
- They are also called as generic pointers.
- They are used when we do not know in advance the data type of the pointer variable.
- They can be made to point to any data type using *type-casting*.
- Syntax of declaration:

```
void *ptr;
```

- Syntax of type-casting:

```
(datatype *) ptr;
```

QUICK EXERCISE

- **Fill in the appropriate format specifiers:**

```
int i=2;
float f=3.5;
void *vp;
vp = &i;
printf( "%__", vp);
printf( "%__", (int*)vp);
printf( "%__", *(int *)vp);
printf ("%__", sizeof (vp));
vp = &f;
printf( "%__", (float*)vp);
printf( "%__", *(float *)vp);
printf ("%__", sizeof (vp));
```

Point to note:

- (datatype *) void_ptr only typecasts the void pointer to the specified datatype.
- In order to access value at a given location an additional "*" is required which is the actual indirection operator.
- size of a void pointer is also 2 bytes like any other pointer variable.

QUICK EXERCISE

- Predict the output (Assume address of i is 715, and f is 915):

```
int i=2;
float f=3.5;
void *vp;
vp = &i;
printf( “%p”,vp);          → 715
printf( “%p”, (int*)vp);   → 715
printf( “%d”, *(int *)vp); → 2
printf (“%d”, sizeof (vp)); → 2
vp = &f;
printf( “%p”, (float*)vp); → 915
printf( “%f”, *(float *)vp); → 3.5
printf (“%d”, sizeof (vp)); → 2
```

CONSTANT POINTERS

- A constant pointer is a pointer that cannot change the address it is holding
- In other words, we can say that once a constant pointer points to a variable then it cannot point to any other variable.
- Note that a constant pointer must also be initialized in the same line of declaration
- Syntax:

```
datatype * const pointer_name = address;
```

- An example declaration would look like :

```
int * const ptr = &x;
```

EXAMPLE

```
int var1 = 0, var2 = 0;
int *const ptr = &var1; ✓
ptr = &var2; X
```

- In the above example :
  - We declared two variables var1 and var2
  - A constant pointer ‘ptr’ was declared and made to point var1
  - Next, ptr is made to point var2.
- If we compile the program, we get an error as follows:
  - constptr.c: In function ‘main’:
  - constptr.c:7: error: assignment of read-only variable ‘ptr’

QUICK EXERCISE

```
#include<stdio.h>
int main()
{
    int a = 5;
    int *ptr ;
    ptr = &a;
    *ptr = *ptr * 3;
    printf("%d", a);
    return 0;
}
```

Output:
15

QUICK EXERCISE

```
#include<stdio.h>
int main()
{
int a = 12;
void *ptr = &a;
printf("%d", *ptr);
return 0;
}
```

Output:
Compiler error.
void pointers must be type-casted before they are de-referenced.

QUICK EXERCISE

Which line will show an error?

```
main()
{
int i=2, j=10;
void  * const vp =&i;
printf("%d",*(int*)vp);
*(int * ) vp=8;
vp = &j;
printf("%d",*(int*)vp);
}
```

Output:
Compiler Error
constant pointers cannot be reinitialized.