EXPERIMENT-8

QUESTION-1: Declare differ hey ent types of pointers(int,float,char) and initialize them with the addresses of variables. Print the values of both the pointers and the variables t point to.

CODE:

```
#include <stdio.h>

int main()
{
int a;
float b;
char c;
printf("enter value of a:");
scanf("%d", &a);
printf("enter value of b:");
scanf("%f", &b);
printf("enter value of c:");
scanf(" %c", &c);

int *p1 = &a;
float *p2 = &b;
char *p3 = &c;

printf("Value of a = %d\n", a);
printf("Address stored in p1 = %p\n", p1);
printf("Value pointed by p1 = %d\n\n", *p1);

printf("Value of b = %.2f\n", b);
printf("Address stored in p2 = %p\n", p2);
printf("Value pointed by p2 = %.2f\n\n", *p2);

printf("Value of c = %c\n", c);
printf("Address stored in p3 = %p\n", p3);
printf("Value pointed by p3 = %c\n", *p3);
```

```
return 0;
}
```

OUTPUT:

(a)enter value of a:45

enter value of b:50

enter value of c:a

Value of a = 45

Address stored in p1 = 0x16eeb3378

Value pointed by p1 = 45

Value of b = 50.00

Address stored in p2 = 0x16eeb3374

Value pointed by p2 = 50.00

Value of c = a

Address stored in p3 = 0x16eeb3373

Value pointed by p3 = a

(b) enter value of a:A

enter value of b:enter value of c:Value of a = 8526736

Address stored in p1 = 0x16f9ef348

Value pointed by p1 = 8526736

Value of b = 0.00

Address stored in p2 = 0x16f9ef344

Value pointed by p2 = 0.00

Value of c = A

Address stored in p3 = 0x16f9ef343

Value pointed by p3 = A

QUESTION-2: Perform pointer arithmetic(increment and decrement) on pointers of different data types. Observe how the memory addresses change and the effects on data access.

CODE:

```c
#include <stdio.h>

int main()
{
int a = 10;
float b = 5.5;
char c = 'A';

int *p1 = &a;
float *p2 = &b;
char *p3 = &c;

printf("Original addresses:\n");
printf("p1 = %p\n", p1);
printf("p2 = %p\n", p2);
printf("p3 = %p\n", p3);

p1++;
p2++;
p3++;

printf("\nAddresses after increment:\n");
printf("p1 = %p\n", p1);
printf("p2 = %p\n", p2);
```

```
printf("p3 = %p\n", p3);

p1--;
p2--;
p3--;

printf("\nAddresses after decrement (back to original):\n");
printf("p1 = %p\n", p1);
printf("p2 = %p\n", p2);
printf("p3 = %p\n", p3);

return 0;
}
```

OUTPUT:

Original addresses:

p1 = 0x16b12b368

p2 = 0x16b12b364

p3 = 0x16b12b363

Addresses after increment:

p1 = 0x16b12b36c

p2 = 0x16b12b368

p3 = 0x16b12b364

Addresses after decrement (back to original):

p1 = 0x16b12b368

p2 = 0x16b12b364

p3 = 0x16b12b363

QUESTION-3: Write a function that accepts pointers as parameters. Pass variables by reference using pointers and modify their values within the function.

CODE:

```c
#include <stdio.h>

void modify(int *x, int *y)
{
*x = *x + 10;
*y = *y * 2;
}

int main()
{
int a = 5;
int b = 6;

printf("Before function call: a = %d, b = %d\n", a, b);

modify(&a, &b);

printf("After function call: a = %d, b = %d\n", a, b);

return 0;
}
```

OUTPUT:

Before function call: a = 5, b = 6

After function call: a = 15, b = 12