

CHAUDHARY HAMDAN

1905387

Networks Lab 1

08/07/2021

1. Write a C program to swap the content of 2 variables using pointer.

Code:

```
#include<stdio.h>

int main() {

    int a = 1, b = 2;
    printf("a = %d, b = %d\n", a, b);

    int *aptr = &a;
    int *bptr = &b;

    int temp = *aptr;
    *aptr = *bptr;
    *bptr = temp;

    printf("a = %d, b = %d\n", a, b);

    return 0;
}
```

Output:

```
a = 1, b = 2
a = 2, b = 1
[Finished in 3.2s]
```

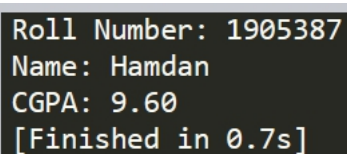
2. Write a C program to assign values to each members of the following structure. Pass the populated structure to a function Using call-by address and print the value of each member of the structure with in that function.

```
struct info{  
int roll_no;  
char name[50];  
float CGPA;  
}
```

Code:

```
#include<stdio.h>  
#include<string.h>  
  
struct info {  
    int roll_no;  
    char name[50];  
    float CGPA;  
};  
  
void print(struct info *stud) {  
    printf("Roll Number: %d\n", stud->roll_no);  
    printf("Name: %s\n", stud->name);  
    printf("CGPA: %.2f\n", stud->CGPA);  
}  
  
int main() {  
  
    struct info stud;  
  
    stud.roll_no = 1905387;  
    strcpy(stud.name , "Hamdan");  
    stud.CGPA = 9.60;  
  
    print(&stud);  
  
    return 0;  
}
```

Output:



```
Roll Number: 1905387  
Name: Hamdan  
CGPA: 9.60  
[Finished in 0.7s]
```

3. Write a C program to extract each byte from a given number and store them in separate character variables and print the content of those variables

Code:

```
#include<stdio.h>

int main() {

    int a = 256;
    int mask = 0xff;
    unsigned char arr[4];

    for (int i = 0; i < 4; i++) {
        arr[i] = a & mask;
        a >>= 8;
    }

    for (int i = 3; i >= 0; i--) {
        printf("%d, ", arr[i]);
    }

    return 0;
}
```

Output:

A screenshot of a terminal window with a dark background. The text '0, 0, 1, 0, [Finished in 0.6s]' is displayed in a light-colored monospaced font. The text is split across two lines: '0, 0, 1, 0,' on the first line and '[Finished in 0.6s]' on the second line.

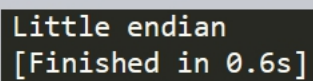
4. Write a C program to check whether the Host machine is in Little Endian or Big Endian.

Code:

```
#include<stdio.h>

int main()
{
    int i = 1;
    char *c = (char*)&i;
    if (*c == 1)
        printf("Little endian\n");
    else
        printf("Big endian\n");
    return 0;
}
```

Output:

A screenshot of a terminal window with a dark background. It displays the output of the C program: "Little endian" on the first line and "[Finished in 0.6s]" on the second line.

```
Little endian
[Finished in 0.6s]
```