```
ubuntu@ubuntu2004:~/Desktop/Ex_10$ ./a.out
Berfore fork p id 8340
Process id= 8340
 Address of var= 76967296
END
Process id= 8343
 Address of var= 76967296
END
ubuntu@ubuntu2004:~/Desktop/Ex_10$ Process id= 8341
 Address of var= 76967296
END
Process id= 8347
 Address of var= 76967296
END
Process id= 8345
 Address of var= 76967296
END
Process id= 8342
 Address of var= 76967296
END
Process id= 8346
 Address of var= 76967296
Process id= 8348
END
 Address of var= 76967296
END
Process id= 8349
 Address of var= 76967296
END
Process id= 8344
Process id= 8350
 Address of var= 76967296
END
 Address of var= 76967296
END
Process id= 8351
 Address of var= 76967296
Process id= 8354
 Address of var= 76967296
END
Process id= 8352
 Address of var= 76967296
END
END
Process id= 8355
Process id= 8353
 Address of var= 76967296
END
 Address of var= 76967296
END
```

```c
int main(){
  printf("Berfore fork p id %d\n",getpid());
  int i=0;
 while(1){
    fork();
     if(i==3)
       break;
     i++;
    }
  int var;
```

rakshit@RG: ~/sample

```c
#include <unistd.h>
#include <sys/types.h>
#include <errno.h>
#include <stdio.h>
#include <sys/wait.h>
#include <stdlib.h>

int main(void)

{ //declare variable

int var=1;

int* p = (int*) malloc(2);
 pid_t PID = fork();
*p = 0;
if (PID >= 0)
{
if (PID ==0)
{

printf("\n\nChild Process:\nInitial Value = %d", var);
var=5;
printf("\nNew Value of var = %d", var);
printf("\nAddress of malloc in child= %p", p);
 printf("\nAddress of var in child= %p\n", &var);
}
else
{
printf("\n\nParent process:\nInitial Value = %d", var);
var = 10;
printf("\nNew Value = %d", var);
printf("\nAddress of malloc in parent= %p",p);
printf("\nAddress of var in child= %p\n", &var);
}
}
return 0;
}
```

```
rakshit@RG:~/sample$ ./exp10


Parent process:
Initial Value = 1
New Value = 10
Address of malloc in parent= 0x55be6a01c2a0
Address of var in child= 0x7ffc6dc3eaf8


Child Process:
Initial Value = 1
New Value of var = 5
Address of malloc in child= 0x55be6a01c2a0
Address of var in child= 0x7ffc6dc3eaf8
rakshit@RG:~/sample$
```