

- 16) write Posix compliant Program that prints Posix defined configuration options supported on given System using feature test macros.

```
#define _POSIX_SOURCE
#define _POSIX_C_SOURCE 199309L
#include "iostream"
#include <unistd.h>
using namespace std;
int main()
{
    #ifdef _POSIX_JOB_CONTROL
        cout << "System Supports Posix Job control;"
            << _POSIX_JOB_CONTROL << endl;
    #else
        cout << "System does not Support Posix
            Job control" << endl;
    #endif

    #ifdef _POSIX_SAVED_IDS
        cout << "System Supports Saved Set UID
            & GID:" << _POSIX_SAVED_IDS;
    #else
        cout << "System does not Support Saved
            Set GID & UID" << endl;
    #endif
}
```



```
# end!f
#!:f def -POSIX -CHOWN -RESTRICTED
    cout << "chown restricted option is:" <<
        -POSIX -CHOWN -RESTRICTED << endl;
# else
    cout << "chown Restricted not defined";
# end!f
#!:f def -POSIX -fio -TRUNC
    cout << "Truncation option is:" << -POSIX -fio
        TRUNC << endl;
# else
    cout << "Truncation not defined" << endl;
# end!f
#!:f def -POSIX -VDISABLE
    cout << "disable char for terminal files "
        << -POSIX -VDISABLE << endl;
# else
    cout << "char for terminal device files will
        not be disabled" << endl;
# end!f
return 0;
}
```