

```
printf("Initializing CULA\n");
status = culaInitialize();
checkStatus(status);

memset(A, 0, M*N*sizeof(float));
err = cudaMemcpy(Ad, A, M*N*sizeof(float), cudaMemcpyHostToDevice);
checkCudaError(err);

printf("Calling culaDeviceSgeqr\n");
status = culaDeviceSgeqr(M, N, Ad, M, TAUd);
checkStatus(status);

err = cudaMemcpy(A, Ad, M*N*sizeof(float), cudaMemcpyDeviceToHost);
checkCudaError(err);
err = cudaMemcpy(TAU, TAUd, N*sizeof(float), cudaMemcpyDeviceToHost);
checkCudaError(err);

printf("Shutting down CULA\n");
culaShutdown();

cudaFree(Ad);
cudaFree(TAUd);
free(A);
free(TAU);

return EXIT_SUCCESS;
}
```