



# Instituto Politécnico Nacional

## ESCOM



### Checksum

Redes

Ortiz Meraz Isaac Baruch

2014081135

30/03/20

## **Checksum o Suma de verificación**

Un checksum, o suma de comprobación, es el resultado de la ejecución de un algoritmo dentro de un archivo único, función denominada Cryptographic hash function. Comparar el checksum que generas desde tu versión del archivo, junto al provisto por la fuente del mismo, representa una ayuda para asegurarte una copia genuina y libre de errores.

Un checksum es también denominado hash sum, hash value, hash code, simplemente hash. Usado típicamente por los algoritmos MD5, SHA-1, SHA-256, and SHA-512.

### **Usos**

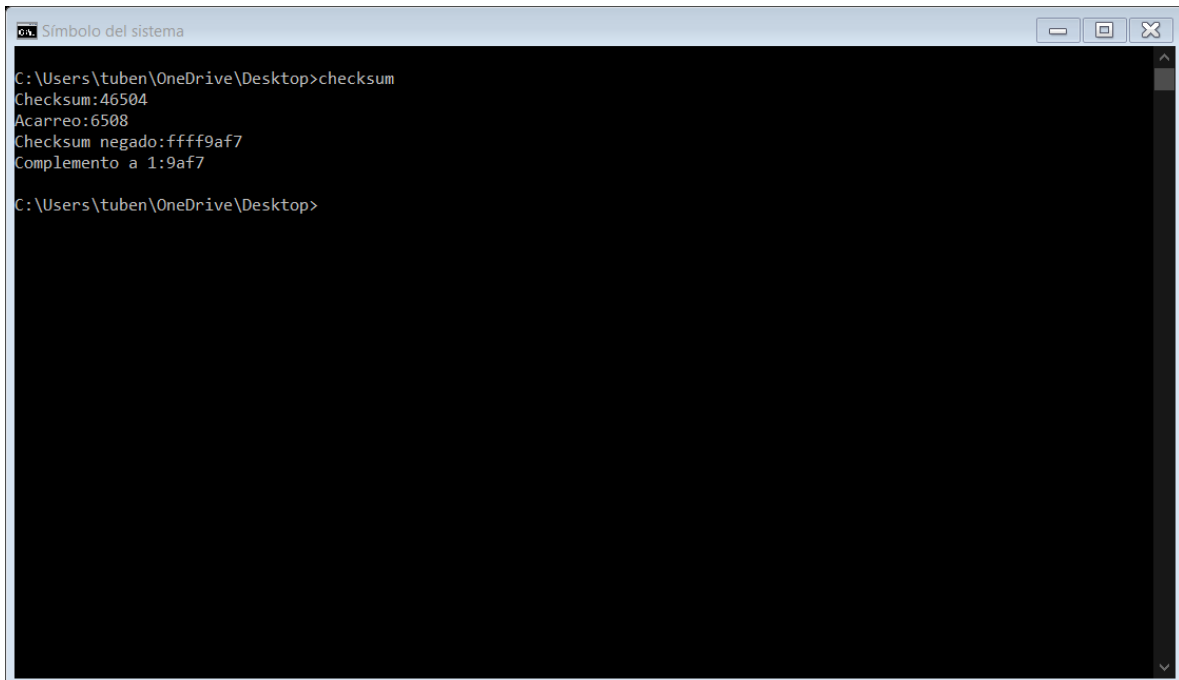
Normalmente se utilizan para verificar los archivos y otros datos, en busca de errores que ocurran durante la transmisión o el almacenamiento. Por ejemplo, un archivo podría no haberse descargado correctamente debido a problemas de red, o los problemas del disco duro podrían haber causado daños en un archivo en el disco.

Si conoce la suma de comprobación del archivo original, puede ejecutar una utilidad de suma de comprobación o hash en él. Si la suma de comprobación resultante coincide, sabrá que el archivo que tiene es idéntico.

Las computadoras usan técnicas de estilo de suma de verificación para verificar los datos en busca de problemas en el fondo, pero también se puede hacer de manera manual.

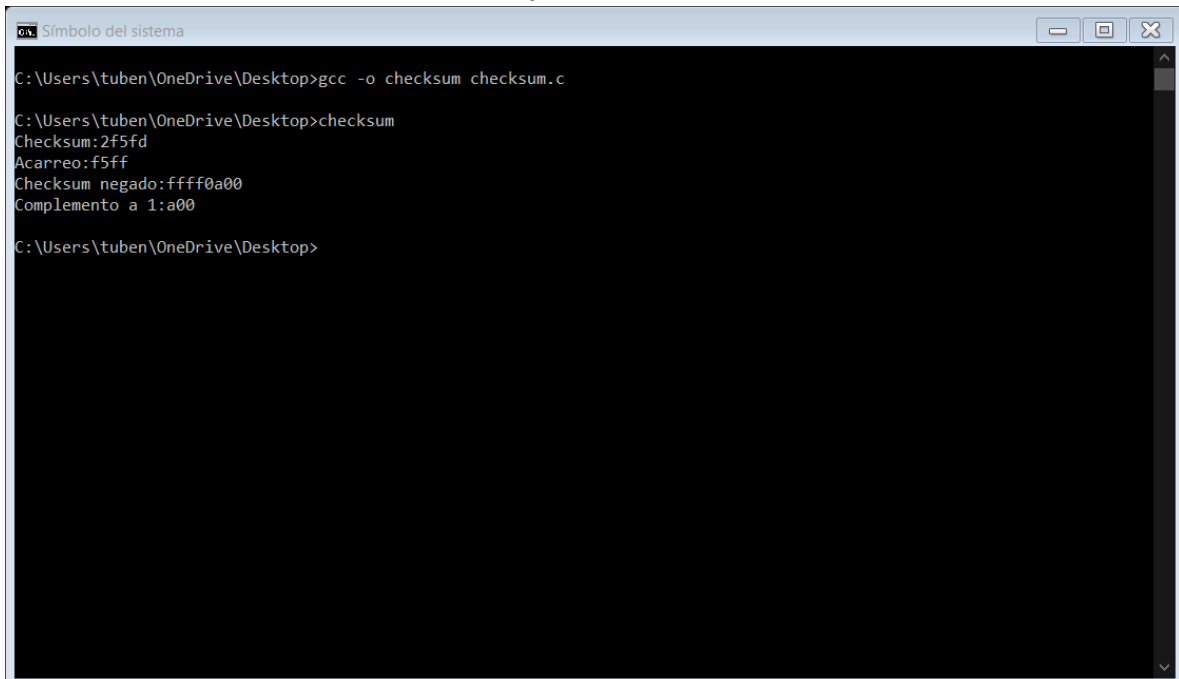
## Tramas de Clase

**tramaprueba[]={0x00, 0x02, 0xb3, 0x9c, 0xae, 0xba, 0x00, 0x02, 0xb3, 0x9c, 0xdf, 0x1b, 0x00, 0x03, 0xf0, 0xf0, 0x7f};**



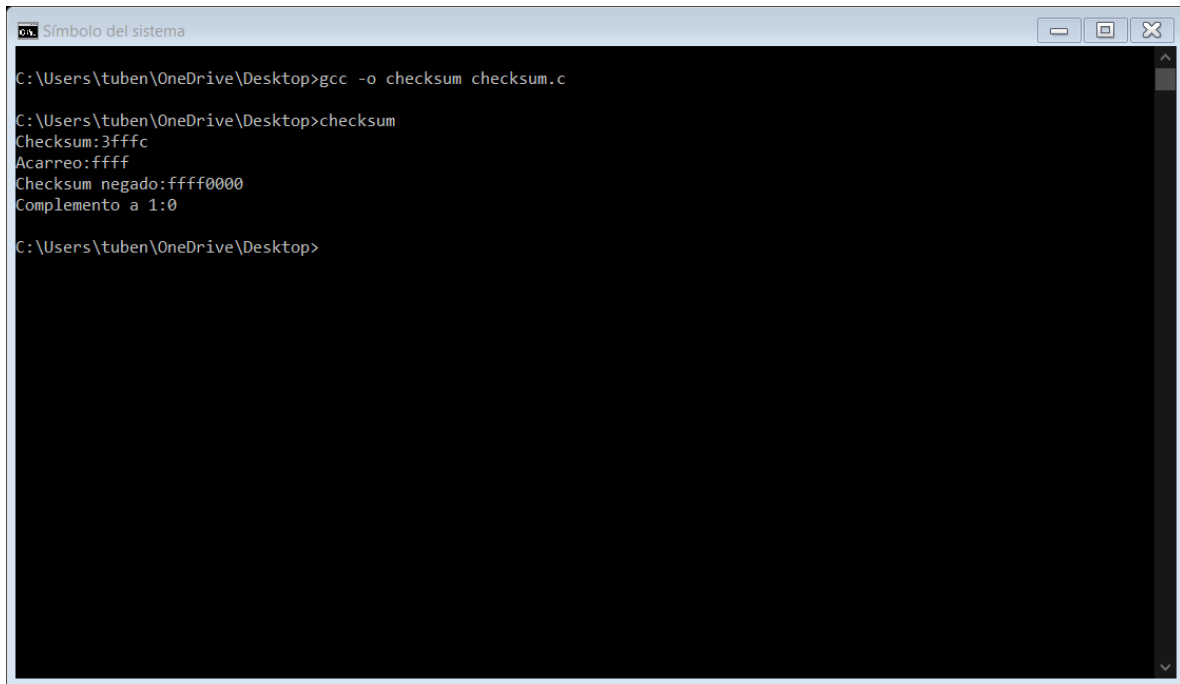
```
Símbolo del sistema
C:\Users\tuben\OneDrive\Desktop>checksum
Checksum:46504
Acarreo:6508
Checksum negado:ffff9af7
Complemento a 1:9af7
C:\Users\tuben\OneDrive\Desktop>
```

**trama1[]={0x45, 0x00, 0x00, 0x3c, 0x04, 0x57, 0x00, 0x00, 0x80, 0x01, 0x98, 0x25, 0x94, 0xcc, 0x39, 0xcb, 0x94, 0xcc, 0x30, 0xe1};**



```
Símbolo del sistema
C:\Users\tuben\OneDrive\Desktop>gcc -o checksum checksum.c
C:\Users\tuben\OneDrive\Desktop>checksum
Checksum:2f5fd
Acarreo:f5ff
Checksum negado:ffff0a00
Complemento a 1:a00
C:\Users\tuben\OneDrive\Desktop>
```

**trama2[]={0x45, 0x00, 0x00, 0x3c, 0x01, 0xd9, 0x00, 0x00, 0xff, 0x01, 0x1c, 0xc8, 0x94, 0xcc, 0xb7, 0xcf, 0x94, 0xcc, 0xbb, 0xb7};**



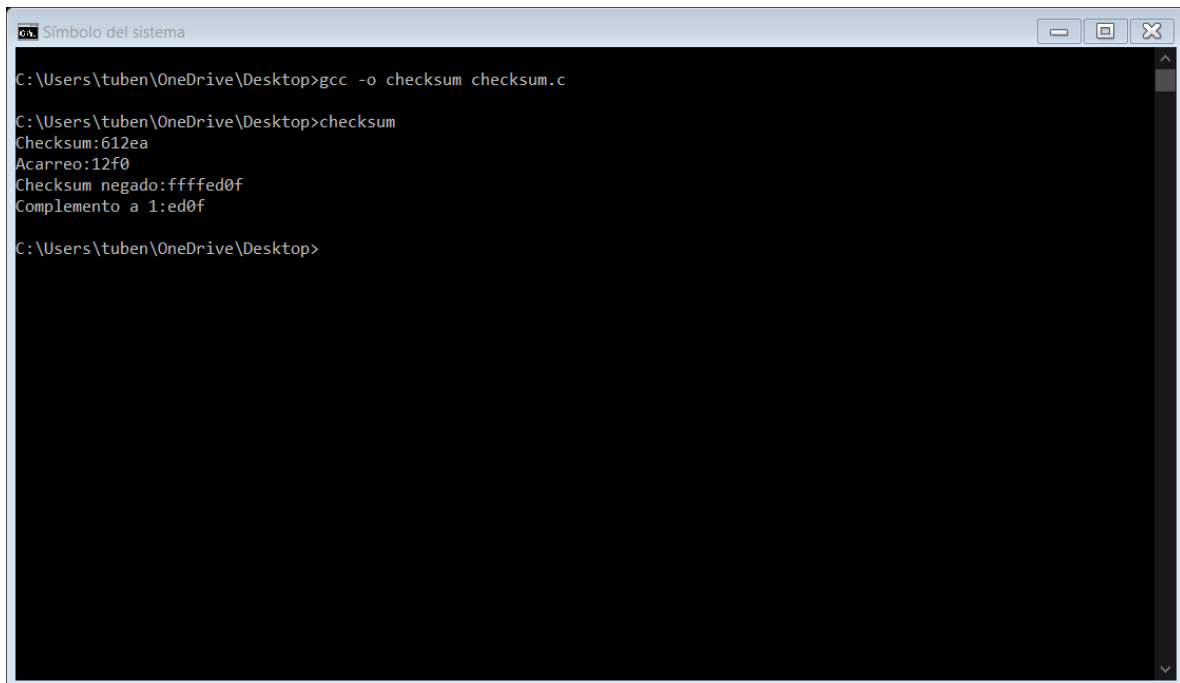
```
Símbolo del sistema

C:\Users\tuben\OneDrive\Desktop>gcc -o checksum checksum.c

C:\Users\tuben\OneDrive\Desktop>checksum
Checksum:3fffc
Acarreo:ffff
Checksum negado:ffff0000
Complemento a 1:0

C:\Users\tuben\OneDrive\Desktop>
```

**trama3[]={0x46, 0x00, 0x80, 0x42, 0x04, 0x55, 0x31, 0x11, 0xb0, 0x11, 0x00, 0x00, 0x94, 0xcc, 0xb7, 0xcf, 0xff, 0xff, 0xaa, 0xbb, 0xcc, 0xdd};**



```
Símbolo del sistema

C:\Users\tuben\OneDrive\Desktop>gcc -o checksum checksum.c

C:\Users\tuben\OneDrive\Desktop>checksum
Checksum:612ea
Acarreo:12f0
Checksum negado:ffffed0f
Complemento a 1:ed0f

C:\Users\tuben\OneDrive\Desktop>
```

## Código

```
#include<stdio.h>
```

```
void checksum (unsigned char trama[],short size){
    unsigned int checksum=0,t,t2;
    short i;
    if(size%2==1){//verifica el tamaño de la trama para saber cuántos va a sumar para
al final sumar la última trama recorrida 8 bits
        for(i=0;i<size-2;i+=2){//suma todas menos la ultima
            t=trama[i]<<8;
            checksum=checksum+t+trama[i+1];
        }
        t=trama[size-1]<<8;
        checksum=checksum+t;//suma la última
        printf("Checksum:%x\n",checksum);
        t=checksum>>16;//aquí se quita el acarreo para sumarlo despues
        t2=checksum&0xFFFF;//aquí deja el numero de la forma 0xFFFF
        printf("Acarreo:%x\n",t+t2);
        checksum=t+t2;//aquí se suman
        checksum=~checksum;//aquí se niega pero queda siempre FFFF y nuestro
numero
        printf("Checksum negado:%x\n",checksum);
        checksum=checksum & 0xFFFF;//aquí quitamos FFFF y dejamos lo demás
        printf("Complemento a 1:%x\n",checksum);
    }else{
        for(i=0;i<size;i+=2){//como el tamaño de la trama es par solo se hace el for
            t=trama[i]<<8;
            checksum=checksum+t+trama[i+1];
```

```

    }

    printf("Checksum:%x\n",checksum);

    t=checksum>>16;

    t2=checksum&0xFFFF;

    printf("Acarreo:%x\n",t+t2);

    checksum=t+t2;//aquí se suman

    checksum=~checksum;//aquí se niega pero queda siempre FFFF y nuestro
numero

    printf("Checksum negado:%x\n",checksum);

    checksum=checksum & 0xFFFF;//aquí quitamos FFFF y dejamos lo demás

    printf("Complemento a 1:%x\n",checksum);

}

}

//Tramas

void main(){

    unsigned char trama[]={0x00, 0x02, 0xb3, 0x9c, 0xae, 0xba, 0x00, 0x02, 0xb3, 0x9c,
0xdf, 0x1b, 0x00, 0x03,0xf0, 0xf0, 0x7f}; //trama prueba

    unsigned char trama1[]={0x45, 0x00, 0x00, 0x3c, 0x04, 0x57, 0x00, 0x00, 0x80,
0x01, 0x98, 0x25, 0x94, 0xcc, 0x39, 0xcb, 0x94, 0xcc, 0x30, 0xe1};

    unsigned char trama2[]={0x45, 0x00, 0x00, 0x3c, 0x01, 0xd9, 0x00, 0x00, 0xff,
0x01, 0x1c, 0xc8, 0x94, 0xcc, 0xb7, 0xcf, 0x94, 0xcc, 0xbb, 0xb7};

    unsigned char trama3[]={0x46, 0x00, 0x80, 0x42, 0x04, 0x55, 0x34, 0x11, 0x80,
0x11, 0x00, 0x00, 0x94, 0xcc, 0x87, 0xcf, 0xff, 0xff, 0xff, 0xff, 0xaa, 0xbb, 0xcc, 0xdd};

    checksum(trama3,sizeof(trama3)); //Para cambiar entre tramas solo se cambia el
numero a un lado de las palabras "trama"

}

```