1. **(25%)** Un número palindrómico es aquel entero que se lee igual al derecho y al revés. Ejemplo: 232 al revés es 232, por ende el 232 es un palindrómico. Realice una función **int espalindromico(int)** que retorne 1 si el número enviado como parámetro es palindrómico y 0 si no lo es.  **No utilizar arreglo para solucionar este problema.**

int espalindromico(int n)

{

int numreves = 0, noria = n;

while ( n != 0 )

{

numreves = numreves \* 10 + n % 10;

n /= 10;

}

if ( numreves == n )

return 1;

return 0;

}

1. **(25%)** Realice una función **int overavg(int valores[], int n)** que recibe como parámetro un arreglo de **n** elementos enteros y retorne la cantidad de elementos que están por encima del promedio.

int overavg(int valores[ ], int n)

{

int cont = 0, ind;

float prom;

for ( ind = 0, prom = 0; ind < n; ind ++ )

prom += valores[ind];

prom /= n;

for ( ind = 0; ind < n; ind ++ )

if ( valores[ind] > prom )

cont ++ ;

return cont;

}

1. **(25%)** Realice una función que determine si una matriz de caracteres se encuentra compuesta únicamente por caracteres numéricos. Defina su función con los argumentos que crea necesarios.

int solonum(char m[][maxcol], int F, int C)

{

int fila, column;

for( fila = 0; fila < F; fila ++ )

for ( column = 0; column < C; column ++ )

if ( m[fila][column] < ‘0’ &&

m[fila][column] > ‘9’ )

return 0;

return 1;

}

1. **(25%)** Realice una función **int cuadruple(char s[])** que recibe como parámetro una cadena de caracteres **s** y debe devolver cuantas palabras de cuatros letras tiene la cadena.

#define TAMPAL 4

int cuadruple( char s[])

{

int ind, inipal, cont = 0;

for ( ind = 0; s[ind]; ind ++ )

{

if ( ind == 0 )

{

if ( s[ind] != ‘ ’ )

inipal = 0;

}

else

{

if ( s[ind – 1] == ‘ ’ && s[ind] != ‘ ’ )

inipal = ind;

}

if ( s[ind] != ‘ ’ && (s[ind+1] == ‘ ’ || s[ind+1] == ‘\0’))

if ( ind – inipal == TAMPAL )

cont ++;

}

return cont;

}