

Centro de Enseñanza Técnica Industrial

Plantel Colomos

Ingeniería en Desarrollo de Software

Nombre Alumno: José Rafael Ruiz Gudiño

Registro: 20110374

Arquitectura de Computadoras

Práctica 2: Suma / resta en punto flotante

3°P

T/M

08/03/2021

Captura del programa

Suma

```
C:\Users\Usuario\Documents\Progra-101\3 Semestre\Arquitectura de computadoras\SinNombre1.exe
                                                                                                                 3-Salir.
Escoge una opcion: 1
Ingresa un numero: 43.75
Ingresa otro numero: .25
43.750000 n1
0.250000 n2
Numero 1 en binario:
101011.11
Numero 2 en binario:
Binario 1 en exponencial:
1.0101111x2^5
Binario 2 en exponencial:
1.x2^-2
Suma:
00000001000000000000000000000000
010110000000000000000000000000000
Resultado suma en Decimal: 44.000000
Process exited after 15.74 seconds with return value 0
Presione una tecla para continuar . . .
```

Resta

```
×
Seleccionar C:\Users\Usuario\Documents\Progra-101\3 Semestre\Arquitectura de computadoras\SinNombre1.exe
1-Suma
2-Resta
3-Salir.
Escoge una opcion: 2
Ingresa un numero: 43.75
Ingresa otro numero: .25
Numero 1 en binario:
101011.11
Numero 2 en binario:
.01
Binario 1 en exponencial:
1.0101111x2^5
Binario 2 en exponencial:
1.x2^-2
Resta:
0000000010000000000000000000000000
0101011100000000000000000000000000
Resultado resta:
Resultado resta en Decimal: 43.500000
Process exited after 10.32 seconds with return value 0
Presione una tecla para continuar . . .
```

Código del Programa

```
#include<stdlib.h>
                                                       printf("Ingresa un numero: ");
                                               scanf("%f",&numf1);
#include<stdio.h>
                                                       printf("Ingresa otro numero: ");
#include<string.h>
                                               scanf("%f",&numf2);
#include<math.h>
                                                       if(numf1<0){ numf1*=-1;} //por si
                                                es negativo
                                                  if(numf2<0){numf2*=-1;}
int main(){
                                                       if(numf1<numf2){
       char
binario1[100],binariof1[100],binariofull1[1
                                                          aux3=numf1;
00],binariofull2[100],resum[100],rerest[10
0],resul1[24],resul2[24]; //cadenas para
                                                          numf1=numf2;
almacenar el primer numero a
                                                          numf2=aux3;
sumar/restar
                                                       }
       char
binario2[100],binariof2[100],bint[24],bif[24
                                                       printf("\n%f n1",numf1);
]; //cadenas para almacenar el segundo
                                                       printf("\n%f n2\n",numf2);
numero a sumar/restar
                                                       n1=numf1;
       int
exp1,longf1,exp2,longf2,cont,temp,long1,
                                                       n2=numf2;
long2,difexp,sumdec,sumint=0,sumf=0;
//para determinar el exponente y longitud
de las cadenas
                                                       numint1= numf1;
                                                                           //Se obtiene
                                               la parte entera
       int
i=0,j=0,numint1,numint2,carry,sum,rest,a
                                                       dcimal1 = (numf1)-numint1;
ux,cont2=0,opc,cero=0; //contadores y
variabla para obtener parte entera
       float numf1,numf2,n1,n2,aux3;
                                                       numint2= numf2;
//Para leer los numeros numero
                                                       dcimal2 = (numf2)-numint2;
       double dcimal1,dcimal2; //Para
obtener la parte fraccionaria
                                                      //CONVERSION DEL SISTEMA
                                                DECIMAL A BINARIO
                                                       //PARTE ENTERA
       printf("\n1-Suma\n2-Resta\n3-
                                                       //Primer numero
Salir.\nEscoge una opcion: ");
scanf("%d",&opc);
                                                       while(numint1>0){ // 5>0// 2>0//
                                                1>0//
              switch(opc){
                     case 1:
```

```
binario1[i]=
                                                         if(dcimal2<1){
numint1\%2;//bin[0] = 1 | bin[1] = 0 | bin[2] =
                                                                 binariof2[j]= 0;
                                                                 j++;
               numint1/=2; //num=2 //
num=1| num=0
                                                         } else{
               ++i; //i=1|i=2|i=3
                                                                 binariof2[j]=1;
       } exp1=i;// exponente
                                                                 dcimal2-=1;
       //PARTE FRACCIONARIA
                                                                 j++;
       while(dcimal1!=0){
                                                         }
       dcimal1*=2;
                                                         } longf2=j;
       if(dcimal1<1){
               binariof1[j]= 0;
                                                    //Se invierte el orden de las cadenas
                                                  de la parte entera
              j++;
                                                    cont=0; temp=0;
       } else{
                                                    while(cont < exp1/2) //invierte el orden
               binariof1[j]=1;
                                                  {
               dcimal1-=1;
                                                    temp = binario1[cont];
              j++;
                                                    binario1[cont] = binario1[exp1 - 1 -
       }
                                                  cont];
       } longf1=j;
                                                    binario1[exp1 - 1 - cont] = temp;
                                                    cont++;
       //Segundo numero
                                                  }
       i=0;j=0; //Se vuelven a igualar a 0
                                                    cont=0; temp=0;
los contadores para utilizarlos
                                                    while(cont < exp2/2) //invierte el orden
       while(numint2>0){
                                                  {
               binario2[i]= numint2%2;
                                                    temp = binario2[cont];
               numint2/=2;
                                                    binario2[cont] = binario2[exp2 - 1 -
               ++i;
                                                  cont];
       } exp2=i;// exponente
                                                    binario2[exp2 - 1 - cont] = temp;
       //PARTE FRACCIONARIA
                                                    cont++;
       while(dcimal2!=0){
                                                  }
       dcimal2*=2;
```

```
//IMPRESION
                                                     for(i=0;i<exp1;++i){}
       //Numero 1
       printf("\nNumero 1 en binario: \n");
                                                          printf("%d",binario1[i]);
                                                          if(i==0) printf(".");
  for(i=0;i<exp1;++i){ //Entero
                                                          if(exp1==0){
       printf("%d",binario1[i]);
                                                          i=0;
                                                          while(binariof1[i]!=1){
       printf(".");
                                                                 if(binariof1[i]==0){
       for(j=0;j<longf1;j++){ //fraccion</pre>
                                                                 cero++;
                                                                 i++;
       printf("%d",binariof1[j]);
                                                                 }
       }
                                                          }
       //Numero 2
       printf("\nNumero 2 en binario: \n");
                                                                 for(i=cero;i<longf1;i++){</pre>
                                                                 printf("%d",binariof1[i]);
                                                                 if(i==cero&&cero<=1)
  for(i=0;i<exp2;++i){ //Entero
                                                  printf(".");
                                                                 }printf("x2^-%d",cero+1);
       printf("%d",binario2[i]);
                                                  cero=0:
       }
                                                          } else{
       printf(".");
       for(j=0;j<longf2;j++){ //fraccion
                                                          for(j=0;j<longf1;j++){
       printf("%d",binariof2[j]);
                                                          printf("%d",binariof1[j]);
}printf("x2^%d",exp1-1); }
       PASAR A MODO EXPONENCIAL
                                                          //NUMERO 2
               printf("\nBinario 1 en
exponencial:\n"); //NUMERO BINARIO
```

EN EXPONENCIAL

```
printf("\nBinario 2 en
exponencial:\n"); //NUMERO BINARIO
                                                    ///concatenar las cadenas
EN EXPONENCIAL
                                                     printf("\nSuma:\n");
  for(i=0;i<exp2;++i){}
                                                    //cadena 1
                                                       /////
        printf("%d",binario2[i]);
                                                       printf("\n");
       if(i==0) printf(".");
                                                       long1 = exp1 + longf1;
        }
                                                       difexp = exp1-exp2;
       if(exp2==0){
                                                       // printf("\ndifexp: %d\n",difexp);
       i=0;
                                                       for(i=1;i<exp1+1;i++){}
        while(binariof2[i]!=1){
                                                            binariofull1[i]=binario1[i-1];
               if(binariof2[i]==0){
                                                            }
               cero++;
                                                            for(i=exp1+1;i<long1+1;i++){
               i++;
                                                            binariofull1[i]=binariof1[i-exp1-1];
               }
                                                            }
       }
                                                            for(i=long1+1;i<33;i++){}
                                                            binariofull1[i]=0;
               for(i=cero;i<longf2;i++){
                                                            }
               printf("%d",binariof2[i]);
                                                            binariofull1[0]=0;
               if(i==cero&&cero<=1)
printf(".");
                                                            for(i=0;i<33;i++){
               }printf("x2^-%d",cero+1);
                                                                    printf("%d",binariofull1[i]);
       } else{
                                                            } printf(" +");
                                                    //cadena 2
        for(j=0;j<longf2;j++){
                                                       long2=exp2+longf2;
                                                       printf("\n");
        printf("%d",binariof2[j]);
  if(exp2==0\&\&j==0\&\&longf2>1)
                                                       for(i=1;i<difexp+1;i++){
printf(".");
               ///por si es 0 el entero
                                                            binariofull2[i]=0;
        }printf("x2^%d",exp2-1);}
                                                            }
        if(n1>=n2){
```

```
for(i=difexp+1;i<exp2+difexp+1;i+
                                                                           case 2: resum[i]=0;
+){
                                                    carry=1; break;
        binariofull2[i]=binario2[i-difexp-1];
                                                                           case 3: resum[i]=1;
                                                    carry=1; break;
        }
                                                                   }}
       for(i=exp2+difexp+1;i<long2+difex
p+1;i++){
                                                                   printf("\n");
               binariofull2[i]=binariof2[i-
                                                            for(i=0;i<32;i++){
exp2-difexp-1];
                                                                   printf("%d",resum[i]);
                                                            }
        for(i=long2+difexp+1;i<33;i++){
                                                           //Se toman los 24 bits de
               binariofull2[i] = 0;
                                                    resultado
        }
                                                            for(i=0;i<24;i++){ //Suma
        binariofull2[0]=0;
                                                                   resul1[i]=resum[i];}
                                                            aux=exp1;
        for(i=0;i<33;i++){
                                                            if(resul1[0]==1) aux++; //por si el
                                                    primer numero es 1 en la suma
               printf("%d",binariofull2[i]);
                                                            if(resul1[0]==0){ //elimina el cero a
        }
                                                    la izquiersa en la suma (si la hay)
        printf("\n");
                                                                   for(i=1;i<24;i++)
        for(i=0;i<33;i++){
                                                                           resul1[i-1]=resul1[i];
               printf("-");
                                                                   }
       }
                                                            }
       i=exp1;
                                                    //imprmir el resultado
       //SUMA
                                                            printf("\nResultado suma:\n");
        carry=0;
                                                            for(i=0;i<24;i++){ //Suma
        for(i=32;i>=0;i--)
                                                                   printf("%d",resul1[i]);
                                                                   if(i==0) printf(".");
        sum=carry+binariofull1[i]+binariof
ull2[i];
                                                            } printf("x2^%d",aux-1);
               switch(sum){
                                                    ///Conversion binario a decimal
                       case 0: resum[i]=0;
                                                            for(i=0;i<exp1;i++){}
carry=0; break;
                                                                   bint[i]=resul1[i];
                       case 1: resum[i]=1;
                                                            }
carry=0; break;
```

```
//
       printf("\nParte entera suma: ");
                                                           if(numf2<0){numf2*=-1;}
//
       for(i=0;i<aux-1;i++){}
                                                             if(numf1<numf2){</pre>
//
              printf("%d",bint[i]);
                                                               aux3=numf1;
//
                                                               numf1=numf2;
//
       printf("\nParte fraccionaria suma:
                                                               numf2=aux3;
.");
                                                        }
       for(i=exp1;i<24;i++){}
                                                        n1=numf1;
              bif[i-exp1]=resul1[i];
                                                        n2=numf2;
       }
                                                        if(numf1<0) numf1*=-1; //por si es
//
       for(i=0;i<24-exp1;i++){
                                                 negativo
//
              printf("%d",bif[i]);
                                                   if(numf2<0) numf2*=-1;
//
                                                        numint1= numf1;
                                                                             //Se obtiene
                                                 la parte entera
       j=aux-1;
                                                        dcimal1 = (numf1)-numint1;
       sumint=0;
                                                        numint2= numf2;
       for(i=0;i<aux;i++){}
                                                        dcimal2 = (numf2)-numint2;
              if(bint[i]==1){cont2=
pow(2,j);} else cont2=0;
                                                        //CONVERSION DEL SISTEMA
              sumint+=cont2;
                                                 DECIMAL A BINARIO
              j--;
                                                        //PARTE ENTERA
       }
                                                        //Primer numero
       printf("\nResultado suma en
Decimal: %f ",n1+n2);
                                                        while(numint1>0){ // 5>0// 2>0//
                                                 1>0//
       fflush(stdin);
                                                                binario1[i]=
                             break;
                                                 numint1\%2;//bin[0] = 1 | bin[1] = 0 | bin[2] =
                                                 1
                     case 2:
                                                                numint1/=2; //num=2 //
          printf("Ingresa un numero: ");
                                                 num=1| num=0
scanf("%f",&numf1);
                                                                ++i; //i=1|i=2|i=3
            printf("Ingresa otro numero:
"); scanf("%f",&numf2);
                                                        } exp1=i;// exponente
              if(numf1<0){ numf1*=-1;}
                                                        //PARTE FRACCIONARIA
//por si es negativo
                                                        while(dcimal1!=0){
```

```
dcimal1*=2;
                                                          } longf2=j;
       if(dcimal1<1){
               binariof1[j]= 0;
                                                     //Se invierte el orden de las cadenas
                                                  de la parte entera
               j++;
                                                     cont=0; temp=0;
       } else{
                                                     while(cont < exp1/2) //invierte el orden
               binariof1[j]=1;
                                                  {
               dcimal1-=1;
                                                     temp = binario1[cont];
               j++;
                                                     binario1[cont] = binario1[exp1 - 1 -
       }
                                                  cont];
       } longf1=j;
                                                     binario1[exp1 - 1 - cont] = temp;
                                                     cont++;
       //Segundo numero
                                                  }
       i=0;j=0; //Se vuelven a igualar a 0
                                                    cont=0; temp=0;
los contadores para utilizarlos
                                                     while(cont < exp2/2) //invierte el orden
       while(numint2>0){
                                                  {
               binario2[i]= numint2%2;
                                                     temp = binario2[cont];
               numint2/=2;
                                                     binario2[cont] = binario2[exp2 - 1 -
               ++i;
                                                  cont];
       } exp2=i;// exponente
                                                     binario2[exp2 - 1 - cont] = temp;
       //PARTE FRACCIONARIA
                                                     cont++;
       while(dcimal2!=0){
                                                  }
       dcimal2*=2;
       if(dcimal2<1){
                                                         //IMPRESION
               binariof2[j]= 0;
                                                          //Numero 1
               j++;
                                                          printf("\nNumero 1 en binario: \n");
       } else{
               binariof2[j]=1;
                                                     for(i=0;i<exp1;++i){ //Entero
               dcimal2-=1;
               j++;
                                                          printf("%d",binario1[i]);
       }
                                                          }
```

```
printf(".");
                                                                 if(binariof1[i]==0){
       for(j=0;j<longf1;j++){ //fraccion
                                                                 cero++;
                                                                 i++;
       printf("%d",binariof1[j]);
                                                                }
                                                         }
       //Numero 2
                                                                 for(i=cero;i<longf1;i++){</pre>
       printf("\nNumero 2 en binario: \n");
                                                                 printf("%d",binariof1[i]);
                                                                 if(i==cero&&cero<=1)
                                                  printf(".");
  for(i=0;i<exp2;++i){ //Entero
                                                                 }printf("x2^-%d",cero+1);
                                                  cero=0;
       printf("%d",binario2[i]);
                                                         } else{
       }
                                                         for(j=0;j<longf1;j++){
       printf(".");
                                                         printf("%d",binariof1[j]);
       for(j=0;j<longf2;j++){ //fraccion
                                                         }printf("x2^%d",exp1-1);}
       printf("%d",binariof2[j]);
                                                         //NUMERO 2
       }
printf("\nBinario 2 en
       PASAR A MODO EXPONENCIAL
                                                  exponencial:\n"); //NUMERO BINARIO
                                                  EN EXPONENCIAL
                                                    for(i=0;i<exp2;++i){}
               printf("\nBinario 1 en
exponencial:\n"); //NUMERO BINARIO
EN EXPONENCIAL
                                                         printf("%d",binario2[i]);
  for(i=0;i<exp1;++i){}
                                                         if(i==0) printf(".");
       printf("%d",binario1[i]);
                                                         if(exp2==0){
       if(i==0) printf(".");
                                                         i=0;
       }
                                                         while(binariof2[i]!=1){
       if(exp1==0){
                                                                 if(binariof2[i]==0){
       i=0;
                                                                 cero++;
       while(binariof1[i]!=1){
```

```
i++;
                                                            binariofull1[i]=binariof1[i-exp1-1];
               }
       }
                                                            for(i=long1+1;i<33;i++){}
               for(i=cero;i<longf2;i++){</pre>
                                                            binariofull1[i]=0;
               printf("%d",binariof2[i]);
               if(i==cero&&cero<=1)
                                                            binariofull1[0]=0;
printf(".");
               }printf("x2^-%d",cero+1);
                                                    //cadena 2
cero=0:
                                                       long2=exp2+longf2;
        } else{
       //if(exp2==0) printf("."); exp2=1;
//////POR SI el exponenete es 0
        for(j=0;j<longf2;j++){
                                                       for(i=1;i< difexp+1;i++){
                                                            binariofull2[i]=0;
  printf("%d",binariof2[j]);
                                                            }
  if(exp2==0\&\&j==0\&\&longf2>1)
                                                            for(i=difexp+1;i<exp2+difexp+1;i+
printf(".");
                                                     +){
                                                            binariofull2[i]=binario2[i-difexp-1];
       }
    printf("x2^%d",exp2-1); }
                                                            for(i=exp2+difexp+1;i<long2+difex
                                                    p+1;i++){
        if(n1>=n2){
                                                                    binariofull2[i]=binariof2[i-
///concatenar las cadenas
                                                    exp2-difexp-1];
//cadena 1
                                                            }
  /////
                                                            for(i=long2+difexp+1;i<33;i++){
  long1 = exp1 + longf1;
                                                                    binariofull2[i] = 0;
  difexp = exp1-exp2;
                                                            }
  // printf("\ndifexp: %d\n",difexp);
                                                            binariofull2[0]=0;
  for(i=1;i<exp1+1;i++){}
                                                            i=exp1;
       binariofull1[i]=binario1[i-1];
                                                    //Resta
        }
                                                            printf("\n\nResta:\n");
       for(i=exp1+1;i<long1+1;i++){}
                                                            for(i=0;i<33;i++){
```

```
printf("%d",binariofull1[i]);
                                                                             case -1:
                                                     if(binariofull1[i]==1&&carry==1&&binariof
        } printf(" -\n");
                                                     ull2[i]==1){
                for(i=0;i<33;i++){}
                                                     rerest[i]=1; carry=1;} else
                printf("%d",binariofull2[i]);
                                                     if(binariofull1[i]==0&&carry==1&&binariof
        }
                                                     ull2[i]==0){
        printf("\n");
                                                     rerest[i]=1; carry=1;} else
        for(i=0;i<33;i++){
                                                     if(binariofull1[i]==0&&carry==0&&binariof
               printf("-");
                                                     ull2[i]==1){
        }
                                                     rerest[i]=1; carry=1;}
        carry=0;
        for(i=32;i>=0;i--)
                                                             break;
                rest=binariofull1[i]-carry-
                                                                             case -2:
binariofull2[i];
                                                     if(binariofull1[i]==0&&carry==1&&binariof
                                                     ull2[i]==1){
                switch(rest){
                       case 0:
                                                     rerest[i]=0; carry=1;}
if(binariofull1[i]==1&&carry==1&&binariof
ull2[i]==0){
                                                                                        break;
                             rerest[i]=0;
                                                                             }
carry=0; } else
if(binariofull1[i]==1&&carry==0&&binariof
                                                                     }
ull2[i]==1){
                                                             printf("\n");
                             rerest[i]=0;
                                                             for(i=0;i<32;i++){
carry=0;} else
if(binariofull1[i]==0&&carry==0&&binariof
                                                                     printf("%d",rerest[i]);
ull2[i]==0){
                                                             }
                             rerest[i]=0;
                                                                     for(i=0;i<24;i++){ //Resta
carry=0;}
                                                                     resul2[i]=rerest[i];
        break;
                                                             }
                        case 1:
                                                     if(resul2[0]==0){ //elimina el cero a la
if(binariofull1[i]==1&&carry==0&&binariof
                                                     izquiersa en la resta (si la hay)
ull2[i]==0){
                                                                     for(i=1;i<24;i++){
rerest[i]=1; carry=0;}
                                                                             resul2[i-1]=resul2[i];
                                                                     }
                                  break:
```

```
}printf("\nResultado resta:\n");
                                                             }
        for(i=0;i<24;i++){ //Resta
                                                     //
                                                             for(i=0;i<24-exp1;i++){
                printf("%d",resul2[i]);
                                                     //
                                                                      printf("%d",bif[i]);
                if(i==0) printf(".");
                                                     //
                                                             }
        } printf("x2^%d",exp1-1);}
                                                             j=aux-1;
                ///Conversion binario a
                                                              sumint=0;
decimal
                                                             for(i=0;i<aux;i++){}
        for(i=0;i<exp1;i++){}
                                                                      if(bint[i]==1){cont2=
                bint[i]=resul1[i];
                                                     pow(2,j); else cont2=0;
        }
                                                                      sumint+=cont2;
                                                                     j--;
//
        printf("\nParte entera resta: ");
                                                             }
//
        for(i=0;i<aux-1;i++){
                                                              printf("\nResultado resta en
                                                      Decimal: %f ",n1-n2);
//
                printf("%d",bint[i]);
//
                                                                                      break;
//
        printf("\nParte fraccionaria resta:
                                                                             default: break;
.");
                                                                     }}
        for(i=exp1;i<24;i++){}
                                                              return 0;}
                bif[i-exp1]=resul1[i];
```

Conclusión

Este programa estaba más complicada que la anterior ya que se tornaba más complicado el hecho de tomar en cuenta las posiciones de las cadenas para de esta manera sumarlos o restarlos según el caso. En mi caso lo más complicado fue la operación de realizar la suma y resta debido a que se tenía que tener mucho cuidado con el acarreo en las operaciones, pero en mi punto de vista fue más complicada realizar la resta ya que tuve que realizar una tabla de verdad para todos los casos posibles en la resta a través de un switch. Aunque en esta práctica estuvo más complicada me gustó y pareció interesante ya que tuve que pensar mucho más. Aparte de que aprendí a sumar y restar en binario y saber lo que se tiene que tomar en cuenta.