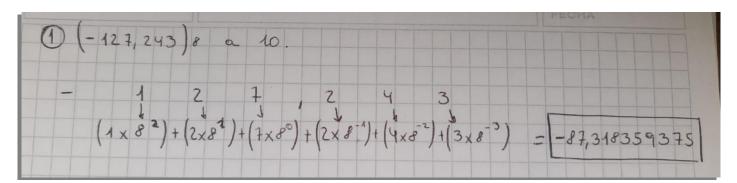
UTN - INSPT 2.601 - Sistemas de Computación I Parcial 2021

Apellido(s), Nombre(s): Alonso Lucio

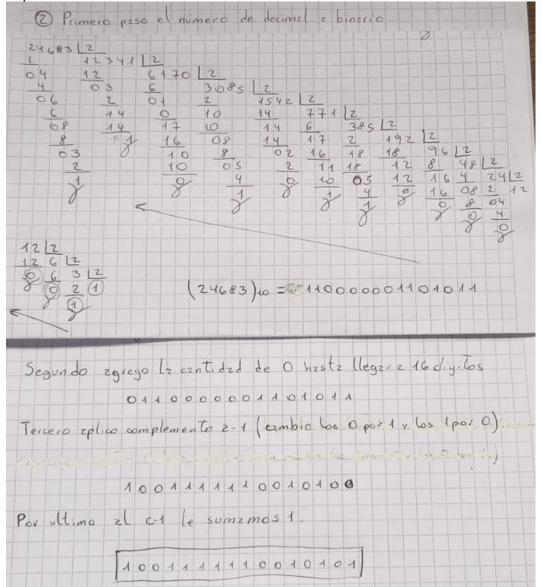
Plantee y resuelva las siguientes cuestiones. Deberá insertar debajo de cada una de las preguntas 1 a 9, claramente legibles de su puño y letra, los pasos seguidos detalladamente y la solución (pueden ser imágenes escaneadas o fotos). La explicación del punto 10 deberá ser de puño y letra y la implementación deberá estar subida en GitLab o GitHub, con el docente invitado a verla como reporter.

El parcial resuelto deberá enviarse, en un archivo PDF, hasta el 09/11/2021 a: corsi@mail.com

1) ¿Cuál es, en decimal, la representación del número octal -127,243?



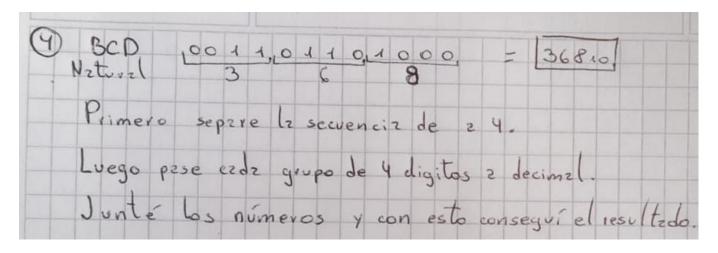
2) ¿Cuáles son, al usar Complemento a 2 (C-2), los 16 bits de la representación del número entero -24683?



3) ¿Cuál es, al usar IEEE 754 (single), el número decimal representado por los 32 bits expresados mediante los 8 dígitos hexadecimales ABAD2020?

3 Primero paso el número Hexadecimal a binario	
(ABAD 2020),=(1010 101110101101001000000000000000000	
1111110000	
2 0100	
1 1 1 9 0000 A 0011 3	
90010 0100 4	
1101 0101 5	27 3
1010	1
0117	
71040 1000 8	
5 15 16 11 + 1001 9	
Segundo Seprio el primer asbit, ya que representa el 1010 A	
el exponente, el rester de les 23 lits sobrantes son la 1100 C	
motiso.	
1110 F	HOLD BY
Poro todos los coguetos a decinal. Liego rezliza el celcula. 1111 F	
5 exponente Matisa	
5 exponente Matisa 1010101110101101001000000000000000	7
S=1 (V=(-1) 5. 2 5-127. 1, M)	1
Exp = 01010111 - 1+2+4+16+64 = 87 -127 = -40	
1 1 2 3 4 5 4 7 9	
Mzt - 010 11010 0 10 0000 0 1 00 0 00 7 1, 01011 0 10 010000000000	
V=(-1) . 2-40. 1,352542877= 1,230130581.10-12 (2+2+2+2+2+2+2+2+2+2+2+2+2+2+2+2+2+2+2	542877

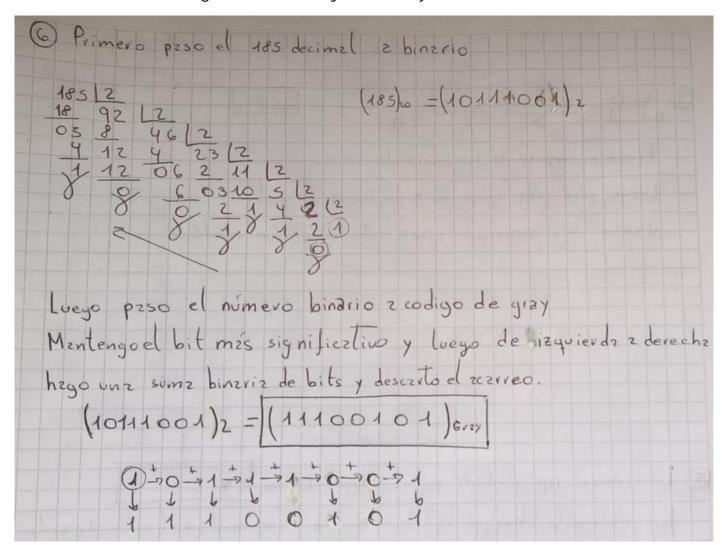
4) ¿Cuál es el número que, al ser representado usando el Código 8421 (BCD natural), da como resultado la secuencia binaria 001101101000?



5) ¿Cuál es, al usar el Código ISO/IEC 8859-1, la cadena representada por los 56 bits 00110101 11010001 01110010 01000101 01100001 00100000 00110001?

Par obtenerlo, busque la tabla donde se encuentra el valor de zada
conjunto de bits.
Loego pase los datos a hexadecimal y braque en latabla elorgeoteraque equivalen. No Bin No Hexa caracter 10011,0101, = 35 = 5
11010001 = D1 = N
2 2 72 5 r
01090104 - 45 - E
6 1
2 0 = Especio
3 1 = 1

6) ¿Cuáles son los 8 bits con los que se representa el número decimal 185 al usar el código binario reflejado de Gray?



7) ¿Cuáles fueron los 7 bits enviados usando el código de Hamming (7, 4), si se recibió la siguiente secuencia binaria con (como máximo) un bit erróneo? 1010110

7 Primero marco los bits de paridad	y de datos
b1 62 b3 b4 b5 b6 b7 [1010110] P1 P2 d1 P3 d2d3 d4	
Verifico la paridad	se necesite un o
P1 = p21id2d (d1, d2, d4) P1=0,0	
p2 = pzridzd (d1,d3,d4) p2=0	,d1=1,d3=1,d4=0
P3 = perided (d2,d3,d4)p3 =0	1, 2=1, 2=1, 24=0
Al zazlizzi, en contremos que en el	bit 1 no hay paridady
quel error se encuentra allí. (Aplicand misma conclusión.	o compuertes XOA rellege el
Corrigien do, el codig o sinal que	

8) ¿Cuántos accesos a RAM se requieren para ejecutar la instrucción POP EBX? Explique y justifique su respuesta

8) De requieren	dos accesos a	le menorie rom. El primero
para desopilar	el dot y el	segent por almorenort en
EBX.		

9) ¿Qué modo de direccionamiento utiliza la instrucción MOV BX, [4691]? Explíquelo basándose en la codificación de la instrucción cuando está alojada en la RAM.

9	El modo de direccionzmiento que emplez la instrucción
	MOV BX, [4691] es el directo.
	Lo que hace es copizr en BX los 16 bits a partir 4691.

10) Explique e implemente las modificaciones requeridas por su compilador de PL/O para que, a partir del código fuente que se muestra a continuación, genere un archivo PE-32 (para Windows) o ELF (para GNU/Linux) que corra según los ejemplos.

```
const MIN = -10000, max=10000;
var A,B,C,W,X,Y,Z;
procedure MOSTRAR;
 IF Y = 0 THEN WRITELN ('HAY UNA RAIZ EN X=', X);
end;
begin
  WRITELN;
  WRITELN ('PROGRAMA PARA BUSCAR RAICES ENTERAS DE ECUACIONES CUADRATICAS');
  WRITELN;
  WRITELN (' 2');
  WRITELN ('A X + B X + C = 0');
  WRITELN;
  write ('Ingrese A: '); readLn (A);
  write ('Ingrese B: '); readLn (b);
  write ('Ingrese C: '); readLn (c);
  WRITELN;
  x := min;
  W := 0;
  while x <= max do
    begin
       z := x - 1;
       y := a * sqr(z+1) + B * x + c;
       if y = 0 then call MOSTRAR;
       x := x + 1
  if W = 0 then writeln ('NO HAY RAICES ENTERAS ENTRE ', MIN, ' Y ', MAX)
```

Ejemplos:

```
2 x^2 - 6 x + 4 = 0

Tiene 2 raíces enteras (x = 1; x = 2)

3 x^2 = 0

Tiene una raíz entera (x = 0)

x^2 - 1 = 0

Tiene 2 raíces enteras (x = -1; x = 1)

2 x^2 + 2 x + 2 = 0

No tiene raíces enteras
```

10 Les modificaciones implementadas en el compilador fueron las siguientes: 1º Se zgregó en la función BLOQUE (dentro del if que detecta un simbolo CONST), que luego de leer un IGUAL (=), puede leerse 2 continueción un NEGATIVO (-) y un Número. Si no entre un regetio, se puscz un número. Código: If ((*simbolo) = - MENOS) Erdenz - t simbolo Neg; Stropy (simbolo Ney, czdenzsimbdo); escenezy (f, restente, simbolo, ezdenz Simbolo, contraos Renglon); If ((*simbolo) == Número) 3 street (simbolo Ney, codene Simbolo); 1 Stropy (czdenz Simbolo, simbolo Ney); 1 table Simb [base + desplazamiento] valor = atoi (como Simbolo) 1 escaneza (firestante, simbolo, cadena Simbolo, contado Renglon) Jelse menszjetri (f, s, czdenz Simbolo); 2º luego se modifico la función búsqueda, ya que esta diferenciaba entre minúsculas y mayúsculas. Esto provocaba que a la hora de quever leer une verieble que habir sido escrita con mayúsculas en su declaración, no se la logrera reconocer como tal si más delante en el có digo se la escribiz en minusculas. Para arreglar esto, se tiens formo, en la compracción de los símbolos, que embos estén en mayusculas usando strupr.

```
Código: Es: 1 (stremp (strupi (nombre Simb), strupi (tiblismb [i] mombre))=0)
              { brezk; }
3º Por último, se zgregó el SQA (pzrz vezlizzr (zpotenciz) el código.
- Primero se colocó dentro del typedef enum el san.
- Segundo, dentro de la función factor, se agrego un cose son, el
 cuzl es el enczygodo de rezlizz la potencia. Detecta la apertura de un
 prientesis, espera una expresión, luego hace los pasajes de memoria correspondentes
 priz rezlieza la potensia del número y por último, espera un cierre de
 prientesis.
Código: cise SQR:
           escaneza (firestinte, simbolo, cadena Simb, contados Renglon);
           1) ((*simbolo) = ABREPARENTESIS)
            escenery (firestrate, ...)
            expression (firestant, simbolo, exdenz Simb, contrdor Renglon)
            czygzyByte (0x5B, memoriz, tope Memoriz); // POP EBX
            czygzr Byte (ox B8, memoriz, tope Memoriz); // MOV EA X X
            czygzylit (OxOO, memoriz, tope Memoriz),
            czygzy Byte ( , memoriz, tope Memoriz), //ADD EAX, EBX
            czigzi Byte ( , memoriz, tope Monoriz),
            czigzi Byte ( , memoriz, tope Memoriz); // IMUL EBX
            czygzy Byte ( , memoriz, tope Memoriz);
            czygzy Byte ( , memoriz, tope Memoriz). Il push EAX
            1) ((*Simbolo) == CIERRAPAMENTESIS) {
escanezy (....)
             felse ...
```

- En le Junción

Godigo: melse if (stremp (erdenz Aux, "san") = = 0)

{* simbolo = san; }

- Yen le Junción imprimir Sambolo se egrego un cese san

Godigo: cese san:

print f ("sann");

breek;