**Preguntas sobre la práctica**

**Ejercicio 1**

|  |  |
| --- | --- |
| **Antes** | **Ahora** |
| **En stack\_fp.h, antes:** | **Ahora:** |
| - | No hay cambios |
| **En stack\_fp.c, antes:** | **Ahora:** |
| En struct \_Pila,  int top; | Elemento \*\*top; |
| En stack\_ini,  top =-1; | top=NULL; |
| En stack\_destroy,  - | while(p->top !=NULL){  element\_destroy(ps->top);  p->top --;  } |
| En stack\_isEmpty  if(p->top == -1) | if(p->top ==NULL) |
| En stack\_isFull  if(p->top == MAX-1) | if(p->top == p-> datos[MAX-1]) |
| En stack\_push  p->top++ | If(p->top ==NULL){  s->top = & (p->datos[0]);  }  Else{  P->top ++;  }  Ps->top = aux; |
| En stack\_pop  p->top --;  return p->item[p->top+1]; | If(p->top != NULL){  Element\_destroy(P->top);  P->top--;  }  Else{  Return NULL;  }  Return p->top; |
| En stack\_top  return (p->item[p->top]); | Return ps->top; |
| En stack\_print  for(i=p->top; i>=0; i--) | aux = p->datos[0];  While(p->top != aux){  num+= element\_print(f, aux);  aux++;  } |
| **En p2\_e3.c, antes:** | **Ahora:** |
| - | No hay cambios |

**Ejercicio 2**

Tenemos que definir un nuevo TAD, el TAD *Aritmetico*, y dentro de él definiremos un operador como:

Typedef struct \_Operador{

char operador;

int prioridad;

};

La prioridad será 0 si el operador es ‘+’ o ’-’, 1 si es ‘\*’ o ’/’ y 2 si es ‘^’.

Dentro de él definimos:

EsOperando(char c)

EsOperador(char c)

ReadFromFile

Prioridad (Operador op1, Operador op2)

En el TAD Element, definimos una función que cree un elemento a partir del operador dado.

Operador\_aEle(Operador op)

No tendríamos que hacer nada con los operandos, ya que el algoritmo utilizado para pasar expresiones infijo a sufijo muestra que los operandos deben ser impresos de inmediato en cuanto se leen. Esto significa que la única pila necesaria para hacer este programa está compuesta por operadores.

Como hemos definido un TAD Aritmetico, podemos prescindir de relacionarlo con el TAD Elemento, ya que ahora podemos meterlo en la pila directamente. Solo tendríamos que crear las tres funciones que se nos piden: copiar, destruir e imprimir un operando.