28/12/23 Weck-2 hab Program - 2 mily write a Program to Convert a given Valid # Paren the sized instruction another the sibrestion to Postjin bebruuin the Estrución Conista of Single Chonalter Oberands & binary Oberators + (Plus), - (minus), + (multiply) / (divide) and 1 (POWER). had (1) total for the work co ! # include < Statio, h> Enter the General # include < Std Cib. h) # include < String . h > # dyine MAX 100 Chan Stack [MAX]; Char infine [MAX]; Chan Possie [MAX]; Int top = - 1; Void push (thon); Char Pop () int is Empty (); void in To Post ()', Void Print (); int Precendence (char)

int mais ()

```
Print & (" enter inject ( sebression : ")
 gets Cinjix);
 into post ();
                            default :
 Print () jodraje -[++i] xidiaog
return 0;
void into Post C)
  int i, J = 0',
 Chan · Symbol next;
for (i=0; i < Strlen Cinjix); i++)
   Symbol = injise [i];
 2
    Switch (Symbol);
    break ,
  (see ') ':
     While (( next = POP()))! = (C')
          Post fix [j++ ]= next;
         break; !! mudso.
   cose '+': thungst
   Cose ( _) : ( went)
   Cose in 1 1
   cose i,,
 While (! is Empty (188 Precendence (Stock > [40])
         >= Precendence (symbol))
```

· c same [++i) signed

```
Postfie G++ ] = POP();
       Push ( Cambol) ; 111 men 1) I tried
       break
    dejant:
          Postbla [j++]= Symbol; () trick
        3
    3
     While Precendence (Comon Symbol)
      6.
          Switch (Csymbol)
        (i=0) i < Staten (inf(x)); i+ 3)
          Cos ' 1'
             return 3;
          Code ( !: (lodings) without
          Cose 'k':
return 2;
                             : 17 / 20
        Cose ! ( 1ct) sist = tran )) still
          Cose 1 1: [] xi7 1809.
            metern 1; jurd
           dejant:
               return 0;
(1 is sundy () so Precendence ( com a school)
```

```
void Print ()
     Print ["The Equivalent Postix Subremiss
              is: " ) ( sort) - strong - )
     pulle (Postfix Ci]) aut - dot
       Print & ("1.c", Possix [i++]);
     Print + ("\n");
   void Oush (charc) 1-== 40+) 41
  Z
      ij (tot = = MAx-1) 3ell
        Print & [" Stack Overflow ");
Enter india Expressión total totales
    a Stacker of top J= c; trulovingor
   2
    Char PopC)
     i) (to ==-1)
```

Print + ("Stack Under (Dow"); but (1); Print & The England Make Edwards C = Stack (top); i top = top - 1([1] x1(, 1809) 1/24 rectuum C; Print ("11.0") Paylex (11+13). int is Empty () Point 6 (" (1)); Void Owh (chore (1-== 40+) Ai oction 1; We (1-x4)/= 401) (i 3 (Company) Starte Orange of Enter injix Expression: axb+c*d-e The Equivalent Postfix Expression is ab " cd " te-

28/12/23 write a Program to Convert a given valid Parentherized infise arithmetre Extración Postix bolb cerion The babrision Consist of Single maractor browneds & binary oboutors and bower write a program: to demonstrate Possisc evaluation: #include <Stdio.h> It include < Stollib.h> # Include < String.h. # dyine MAX_SIZE 7100 (117 18) topins int Stack EMAX (SIREJ) (1) Mas) Many int top = -1; Void Push (int Element) if Ctob > = MAX STEED Down 1/11 S int operating 1 = 1000 3 Print & ("Stack Ovaylow") oction 3 3 Elec & Stack [++top] = Element; James 10 Z 3 int pop 12 & 1) (top <0) { Print & (" Stock Underflow \n"); return -1', the & Stack

Lab Program - 3

```
a wind of morning
  int Evaluate Postfix (char # sub) 2
   int len = Str len (Esch);
     tor (i=0; i < tun; i++)
    2
     if
    Cis digit Cru [i]) I Jare - >(2)
       Push (Edt [i] [ionia) your
                 (int slewest)
        Elec &
      int Oborator Operand 2 = POP();
      int operand 1 = Pop(); 5 you)
     Switch (Expritighter of shop ")
     2
         Cose 't';
 Push (oberano(1+ operanol2);
   proof ;
     Cose 1-1.
  Pish (Oberand 1 - Oberand 2).
    break;
 Cose 1 x1.
Push (obcirand 1 + oterand 2);
  Core 1/ 1.
```

puph (operand: /Orerand 2); break of annual of marginal June to dimension in a comment is the material the or and in the rout pope); return rout; " () Jenni biox () White Liev Condition how chor Ext [Max - 9'ac]; Orint L'Enter the Postfix Extremión : 1700 Scant ("1.1.5", bub); intres = Evaluationostix (Exb); Print + C'Reeut = 1/d \n 1, rcs)(1) Print ! (" insert Elevent to Buse !:"). the Postix Education: 1 11119 Tring (11 Disting an Elevante of Buch Che Derut : 11 Prints (" Built In"); Some f (" Enter your chair (de) Scart (419 4 & choice); Sunder (chalce)

```
Week-3 more 1 seros to) way
          Program - 3
> horite a Program to devorstrate
> NAP to Simulate working of a Queue of
    Integors wing an averay.
It include estato, h>
# define MAX 50
  void insert();
  void delde ();
 void destroyer;
 Int Oucue away CMAXJ:
  int rear = -1;
 int bont = alignated xipped at many of the
 main ()
                    Sent ("15" wb);
  ર્
              INTERES = . English (POSH) = ((EN b))
    int choice;
    while (1) 25 - 1 - d /n o mile (1) I mile
    1
     Print F (" insert Element to Ducue In"):
     Print of (" insert Delete Element from Queux m").
    Print (" Distloy all Elemente of Queuc In");
  Printe (" Quit In");
  Print f ( " Enter your choice");
  Scant (u.r.d", & choice);
  Switch (choice)
   Cose 1 %
    inscrtC);
   bocex;
```

```
( ) -b lata they
   Cose 2:
    dutas;
    preak;
    Cose 3:
    Print (" Ourus Orderfess / n. F. C. gradail
   break;
   Cose 4:
    Suit (1);
   dejault:
   Print + ('brong choice \n').
Busic is 1.d / h. " Ourine_cooger & D.
void insort ()
                             ( ) partob biox
  int add - item;
  if (rear = = MAX -1)
  Print & ("Que de Droylow \n"); il Ini
                      (1-== + nos+ ) 71
        Print P (" Quelle c in Grantly (n');
 Elsc
  if (front = = -1)
  Print f ("insert the Element in Queble !!);
  front = 0.
  Swanf (""/d", Sadd_item);
                  Print + (w/d" Emeric
  7 con = 7 con + 1;
 Onene_avong Creon J= add, -item;
```

```
duur ()
Void
 it (bont == -1 11 front > rear)
 Ş
   Printf (" Oucue Underflow (n");
    return;
Elsc
S
 Print ("Element deletal from
   Aueuc is: 1.d \n: Oucuc_covaychont].
     front = front fl;
   ٤
f
 void distray ()
ર
   int i; ("K/ wayoud all we?") + min
  if (front==-1)
    Print & Cu Queble is Empty (n");
 Else
                       ( ) - = = 1 mont)
 Print F. ("Ouçue is: \n");
  for (i= foont; i <= rear; i++)
    Print + ["1.d", Ducue _avong ci ]);
   Print & C"\n"); = [ som
3
```

output insert blement to Quene Delide Element from aucui hand 3. distribuy all believent of Quinciples 14 a aint Enter your choice: 1 munt the Meinert of Ducce : 42 enter your choice si 3 Quee if 42 Enter your above? 2 Element deleted from freeze et 42 Enter your choice (1-==) next) - i Su'b Void En Busine (Int Exercises)] But the (" /10 Queence is full 1 ("): , 35B / (1+ coop). tusos Him Erward - Eleveriti.

```
Include <Stato.n>
define MAX 3
oid insert();
oid delete();
oid display();
nt queue_array[MAX];
nt rear = - 1:
nt front = -1:
ain()
   int choice;
   while (1)
   {
       printf("1.Insert element to queue \n");
       printf("2.Delete element from queue \n");
       printf("3.Display all elements of queue \n");
       printf("4.Quit \n");
       printf("Enter your choice : ");
       scanf("%d", &choice);
       switch (choice)
       {
           case 1:
           insert();
           break:
           case 2:
           delete();
           break:
           case 3:
           display();
           break:
           case 4:
           exit(1);
           default:
           printf("Wrong choice \n");
       }
  }
oid insert()
   int add_item:
   if (rear == MAX - 1)
   printf("Queue Overflow \n");
   else
   {
       if (front == - 1)
```

```
oid insert()
   int add_item;
  if (rear == MAX - 1)
  printf("Queue Overflow \n");
   else
       if (front == - 1)
       front = 0:
       printf("Inset the element in queue : ");
       scanf("%d", &add_item);
       rear = rear + 1;
       queue_array[rear] = add_item;
oid delete()
  if (front == - 1 || front > rear)
       printf("Queue Underflow \n");
       return ;
  else
       printf("Element deleted from queue is : %d\n", queue_array[front]);
      front = front + 1;
   }
oid display()
   int i:
  if (front == -1)
       printf("Queue is empty \n");
   else
       printf("Queue is : \n");
       for (i = front; i <= rear; i++)
           printf("%d ", queue_array[i]);
       printf("\n");
   }
```

```
Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 1
Inset the element in queue : 10

    Insert element to queue

Delete element from queue
Display all elements of queue
4.Quit
Enter your choice : 2
Element deleted from queue is : 10

    Insert element to queue

Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 20
Wrong choice

    Insert element to queue

Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 2
Queue Underflow

    Insert element to queue

Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 3
Queue is :

    Insert element to queue

Delete element from queue
3.Display all elements of queue
4.Quit
Enter your choice : 4
Process returned 1 (0x1)
                          execution time : 17.302 s
Press any key to continue.
```

Insert element to queue

```
#include<ctype.h>
char stack[100];
int top = -1;
void push(char x) {
   stack[++top] = x;
char pop() {
   if (top == -1) {
       return -1;
   } else {
       return stack[top--];
   }
int priority(char x) {
   if (x == '(')
       return 0:
   if (x == '+' || x == '-')
       return 1;
   if (x == '*' || x == '/')
       return 2;
   return -1;
int main() {
   char exp[100];
   char *e, x;
   printf("Enter the expression: ");
   scanf("%s", exp);
   e = exp:
   while (*e != '\0') {
       if (isalnum(*e)) {
           printf("%c", *e);
       } else if (*e == '(') {
           push(*e);
       } else if (*e == ')') {
           while ((x = pop()) != '(') {
               printf("%c", x);
       } else {
           while (ton !- -1 && priority(etack[ton]) >- priority(ta)) {
```

```
nt priority(char x) {
   if (x == '(')
       return 0;
   if (x == '+' || x == '-')
       return 1;
   if (x == '*' || x == '/')
       return 2;
   return -1:
nt main() {
   char exp[100];
   char *e, x;
   printf("Enter the expression: ");
   scanf("%s", exp);
   e = exp;
   while (*e != '\0') {
       if (isalnum(*e)) {
           printf("%c", *e);
       } else if (*e == '(') {
           push(*e);
       } else if (*e == ')') {
           while ((x = pop()) != '(') {
               printf("%c", x);
       } else {
           while (top != -1 && priority(stack[top]) >= priority(*e)) {
               printf("%c", pop());
           }
           push(*e);
       }
       e++;
   }
   while (top != -1) {
       printf("%c", pop());
   }
   return 0;
```

```
nter the expression: a*b+c*d-e
ab*cd*+e-
Process returned 0 (0x0) execution time : 10.191 s
Press any key to continue.
```

```
int stack[3];
int top = -1;
void push(int x)
    stack[++top] = x;
int pop()
    return stack[top--];
int main()
    char exp[20];
    char *e;
    int n1, n2, n3, num;
    printf("Enter the expression :: ");
    scanf("%s",exp);
    e = exp;
    while(*e != '\0')
    {
        if(isdigit(*e))
            num = *e - 48;
            push(num);
        }
        else
        {
            n1 = pop();
            n2 = pop();
            switch(*e)
            {
            case '+':
            {
                n3 = n1 + n2;
                break;
            }
            case '-':
                n3 = n2 - n1;
                break;
            case '*':
            {
                n3 = n1 * n2;
```

{

}

{

```
int n1, n2, n3, num;
printf("Enter the expression :: ");
scanf("%s",exp);
e = exp;
while(*e != '\0')
    if(isdigit(*e))
        num = *e - 48;
        push(num);
    else
        n1 = pop():
        n2 = pop():
        switch(*e)
        case '+':
            n3 = n1 + n2;
            break;
        case '-':
            n3 = n2 - n1;
            break;
        case '*':
            n3 = n1 * n2;
            break:
        case '/':
            n3 = n2 / n1;
            break:
        push(n3);
    e++:
printf("\nThe result of expression %s = %d\n\n",exp,pop());
return 0;
```

nter the expression :: 23*5+

he result of expression 23*5+ = 11

rocess returned 0 (0x0) execution time : 8.538 s

ress any key to continue.