```
C. Program to implement STACK using arrays/global
# include < stdio . h >
# include < process. h>
# include / conio.h>
# define STACK_SIZE 5
int top = -1;
 int s[10];
 int item;
 roid push ()
 4 (top == STACK-SIZE-1)
  2 printf ("stack overflow \n");
  return;
  top = top+1;
  5[top] = "item";
int pop()
  if (top = = -1) return -1;
  return s[top--];
 wid display ()
至此声
   if (top = = -1)
   ¿ printf ("stack is empty \n");
    return;
   print ("contents of the stack \n");
    for ( i=0; i <= top; i++)
     2 print ("Y.d \n", s[i]);
```

```
wid main ()
à int item-deleted;
   int droke;
    elser ();
    for (; ;)
    print ("In 1: puch In 2: pop In 3. display In 4: exit In");
     printf ("enter the choice \n");
     scarf ("%d", & choice);
   surter (choice)
   case 1° point ("enter the "tem to be inserted In");
          scarf ("% d", & item);
          push ();
          break;
   rase 2: item - deleted = pop ();
           "4 (item_deleted == -1)
            privily ("stack is empty \n");
           print ("item deleted is %d \n", item-deleted);
           break;
   case 3: display ();
           break;
    default: exit (b);
  getch();
```

```
C-Program to implement STACK using arrays pointer
# include = stdio. h >
It include & process-ho
# wirelude < como. h>
# define STACK_SIZE 5
int top = -1;
wid push ( int item, int sty, int * top)
   if ( top == STACK_SIZE - 1)
     print ("stack overflow(n");
   5[*top]= item;
  int pop(int s[], int *top)
    int item-deleted;
    if (*top====1)
    E print ("stack or underflow cannot delete (n');
     return 0;
    iten_deleted = 5[*top];
   *top = *top-1;
neturn êtem-deleted;
   roid display (int top, int s[])
      "4 (top == -1)
      & printf ("stack is empty \n");
      printf ("contents of the stack \n");
```

```
for li=0; iz=top; in)
   printf ("ofad In", s[i]);
soil main ()
  int item, s[10];
   int item-deleted;
   int whoice;
   closer();
   for ( ;;)
    print ("\n1: push\n2: pop\n3: duplay \n4".exit \n");
     printf ("enter the choice \n");
     scarf ("10d", & choice);
     builth (choice)
   case 1: point ("enter the item to be inserted \n');
              push (item-deleted 6 = 0) push (item, S. & top);

priority ("item deleted is "/ed \n", item-deleted);
              scarf ("% d", & item);
   case 2: "item-deleted = pop (s, & top);
               if ("item - deleted ! = 0)
              print (" item deleted is % od In", item-deleted);
              break;
    core 3% display (top, s);
             break ;
     default : exit(0);
   3 gatch ();
```