

5. no. of Character/line P. vijayvardhan Reddy
Reg no: 192011179

Aim: To write a C program for implementing a lexical Analyzer to scan and count the numbers of character's words & lines in a file.

Program:-

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    file <file>;
    char ch;
    int character; words; lines;
    printf("Enter source file path");
    scanf("%s", path);
    file = fopen(path, "r");
    if (file == NULL)
        printf("I'm unable to open file exit")
        printf("Please check if file exists & you have read\n");
        exit(EXIT_FAILURE);
    }
    character = words = lines = 0;
    while (ch = fgetc(file) != EOF) {
        character++;
        if (ch == ' ' || ch == '\t' || ch == '\n' || ch == '\0')
            words++;
    }
}
```

```

if (characters > 0) {
    words++;
    lines++;
}

```

P. Vishnu Varshan Reddy
Reg no: 19201179.

```

printf("\n");
printf("Total character = %d\n", characters);
printf("Total words = %d\n", words);
printf("Total lines = %d\n", lines);
return 0;
}

```

output:-

Enter source file path input.txt

Total character = 8.

Total words = 2.

Total lines = 1.

Result:- C program for implements a lexical

Analysis to scan & count the no. of characters, words & lines in a file.

Ans

6. Computation of first: 192011179.

Aim: To write a C program for a find first & follow for the Predictive Parser.

Program:

```
#include <stdio.h>
#include <type.h>
void find (char *char);
void add to result set (char *char);
if num of productions;
char production set [10] [10];
int main ()
{
    int i;
    char choice;
    char c;
    char result [20];
    printf ("How many no. of production");
    scanf ("%d", &i);
    do
    {
        printf ("Find the first of:");
        scanf ("%d", &choice);
    }
}
```

5. void list(char * result[20]);

P. V. M. N. S.
1920/11/17/9.

Int found Eposition;

$$\text{sub Result}[0] = 1101 =$$

Result [0] = '10'

return;

3.

```
int k;
```

```
int k;  
for(k=0; Result(k) != 0; k++).
```

```

for (k=0, result = 0)
    if (Result[k] == val)

```

return;

```
Result[k] = Val;
```

Result (1) $\rightarrow 16$;

3.

output:-

output:-
 Enter production 1 $M1 = F = 9D$
 $P = 7TD$

Ende production $m_2 = D - 7TD$

~~Enter production $\mu_3 = 77$~~

Enkr production $mu = 5 \pm 8$

Enter production ms = 5 : 8

Result:- C program for computation of TET
is Executed successfully.

in Executed successfully.

7. Computation of Follow

Aim:- Write a C program to find follow set for Predictive parser.

Program:-

```
#include <stdio.h>
#include <type.h>
char production[10][10]; array[10],
int main()
{
    int count;
    char option; ch;
    printf("Enter total no. of production");
    scanf("%d", &count);
    for (count = 0; count < limit; count++);
    do
    {
        printf("Enter production value to find follow");
        scanf("%d", &ch);
        void find_follow (char);
        {
            limit i, j;
            int length = strlen (production[i]);
            if (production[0][0] = ch)
```

void findAns (char ch)

{
if (ch == 'A' || ch == 'a') {

 // empty population (ch)

 if (production[1][0] == 0)

 {
 // follow (production[1][0]);
 }

else if (isLower(production[1][0]))

{
 //

 // find - first (production[1][0]);
}

output:-

Enter total no. of production test

Follow case of D.

To continue press 'n'.

Result:- Program for find follow is

successfully completed.

8. COMMENT OR NOT (P. VISHNU 20)

Aim:- To Implement a program to identify whether a given line is comment or not.

Program:-

```
#include <stdio.h>
#include <conio.h>
void main()
{
    char com[30];
    int i=2, a=0;
    printf("Enter comment");
    a=1;
    break;
    if (a==0)
        printf("It is not a comment");
    }
    else {
        printf("It is comment");
    }
}
```

Output:-

Enter comment hello
It is comment.

Result:- C program to identify whether a given line is comment or not is completed.

9. Identifier or not? P. Vithnu 2/1
19/01/179.

Aim! write a c program to test whether given identifier is valid or not.

Program!

```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
void main()
{
    char a[10];
    int flag; i=1;
    clr();
    printf("Enter a identifier:");
    gets(a);
    printf "is alpha[a(0)].
    flag=1; else.
    printf("Not valid");
    while (a[i]!='\0').
    { flag=0; break.
    } getch();
}
```

output: Amt.
valid identifier.

Result: The C program for identifier successfully completed.

Ans 25

how to know if you're

Strophiles humilis

Handwritten text: *Handwritten text, possibly a signature or name, in cursive script.*

Ch. 9742-96, 10000 H

1. Stress is not the same as anxiety

1. Goal and end

John Paul Jones

10.15

Conductivity or resistivity

$$f(n) \, d\mu$$

(a) $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7, \alpha_8, \alpha_9, \alpha_{10}$

9129, 1-12-19

(7) Limited - small

$$\cdot (b) = (c) T \Delta S_{\text{mix}}$$

Handwritten: 1/20/19

154

131

Vol. Mohi Lila

Aluminum

1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 26

6/19/94 (cont) 10/1/94

break;

19/11/20

switch (str):

```
{
case '1':
str = 0;
break;
case '4':
str = 1;
case 'K':
str = 2;
break;
case 'c':
str = 3;
break;
case '5':
str = 4;
break;
case 'x':
str = 5;
break;
}
```

printf ("n success");

getch();

}
 output:
 xbt jks
 x b8 success

Result:- C program for constructing LL is completed.

Ans 8/23

11. Construct Recursive Descent Parsing 24

P.V. Vinnu.
19/01/18-9.

Aim: To Execute a program to construct recursive descent parsing.

Program:

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
char input[100];
int i, 1;
void main()
{
    if (input[i] == '\0')
        printf("String is accepted");
    else
        printf("String is not accepted");
    getch();
    else
        return (0);
    EP()
    {
        if (input[i] == '+')
        {
            H+, if T()
        }
        {
            if (EP())
                return (1);
            else
            {

```

P. V. Thirumala
1920/11/8-9

Answers

his 7-6000 ft.

1944

4403

$\frac{d}{dt} \left(\frac{1}{r^2} \right) = -\frac{2}{r^3} \frac{dr}{dt}$

1. The first part of the paper is devoted to a discussion of the

Aim: To write a C program for stack implementation to the shift reduce parser.

Program:

```
#include <stdio.h>
#include <stdlib.h>
char ip, sym[5], stack[5];
char act[15];
int main() {
    printf("Shift Reduce Parser");
    printf("\n Grammar");
    len = strlen(ip + sym);
    ip - sym[ip - ptr] = '\0';
    ip - ptr = 0;
    str = ptr;
    str - ptr++;
    check(1);
    if flag == 0)
        exit(0);
    return;
}
// a b c = x a b c d - -
// a b c . shift.
```

Result:

C program for stack shift reduce parser executed.

13. Operation Precedence Parser. Prithvi 27 19201179.

Aim: To write a C Program for operation Precedence Parsing.

Program:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
char *input; int i = 0;
```

```
char *lathandle, *stack[50], *handle[3][5];
```

```
top = 0;
```

```
char Pre[9][9] =
```

```
switch (C)
```

```
{
```

```
case '+':
```

```
return 0;
```

```
case '-':
```

```
return 1;
```

```
case '*':
```

```
return 2;
```

```
case '/':
```

```
return 3;
```

```
}
```

```
return 0;
```

```
}
```

O/P:

if * (1+1) as shift.
OR * (1+1) as Reduced;

Result:

C Program for operation Precedence Parsing
Executed successfully.