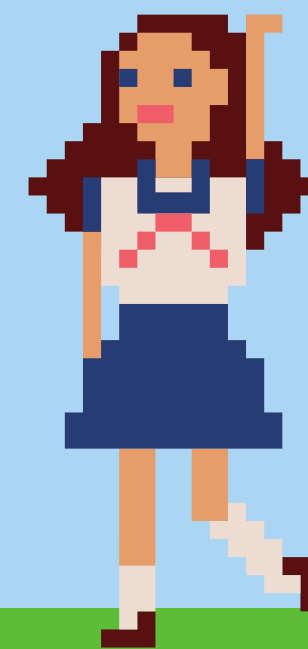
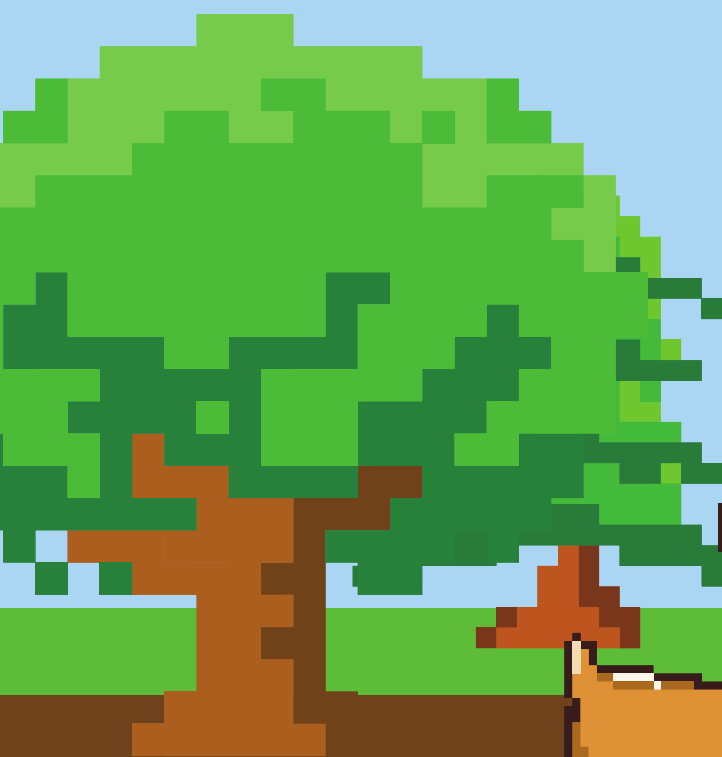


# CALCULATE POSTFIX STACK LINKED LIST

Presented by: เขียนโค้ดแบบงูดลาด





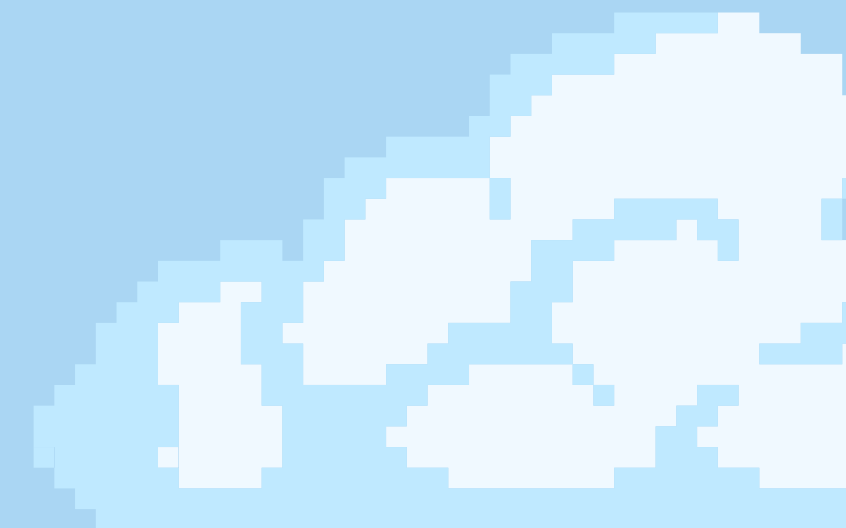
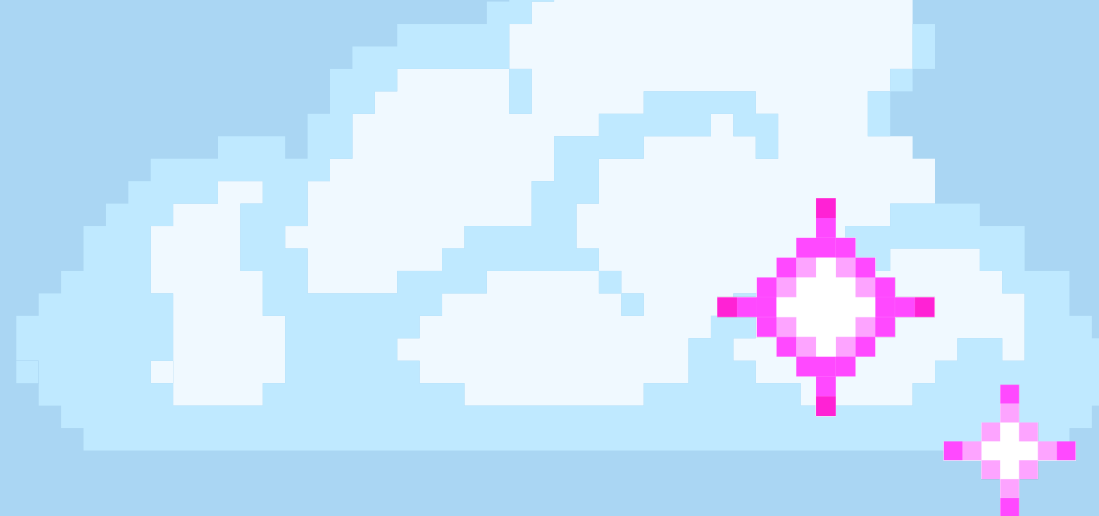
## ประกาศ STRUCTER

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <stdbool.h>
5
6  typedef struct  s_stack
7  {
8      int          data;
9      struct s_stack *next;
10 }  t_stack;
```

## FUNCTION PUSH

```
1  // create and add linked list to first of stack
2  bool    push(int num, t_stack **stack)
3  {
4      t_stack *dest;
5
6      dest = NULL;
7      dest = (t_stack*) calloc (sizeof(t_stack), 1);
8      if (!dest)
9          return (0);
10     dest->data = num;
11     dest->next = *stack;
12     *stack = dest;
13     return (1);
14 }
```

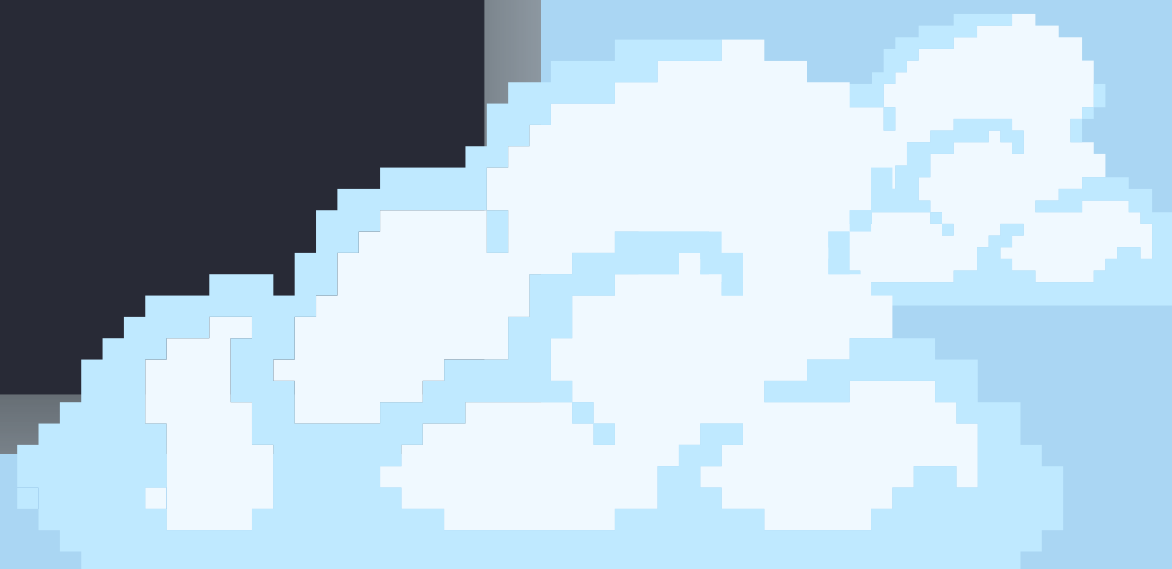


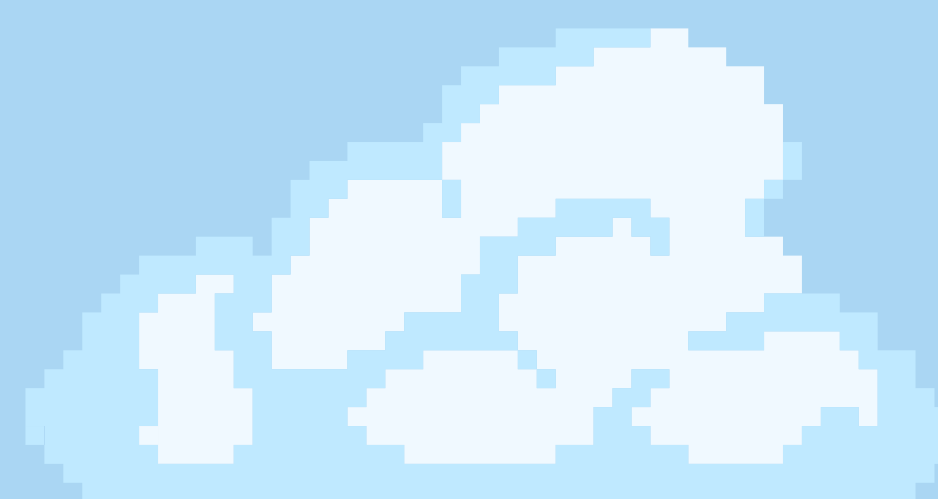


```
1 // pop first linked list of stack
2 bool    pop(int *top, t_stack **stack)
3 {
4     t_stack *temp;
5
6     // if stack is not NULL then pop!
7     if (*stack)
8     {
9         temp = *stack;
10        *top = temp->data;
11        *stack = (*stack)->next;
12        free(temp);
13        return (1);
14    }
15    else
16        return (0);
17 }
18
```

## FUNCTION POP

การ POP ข้อมูลออกจาก STACK



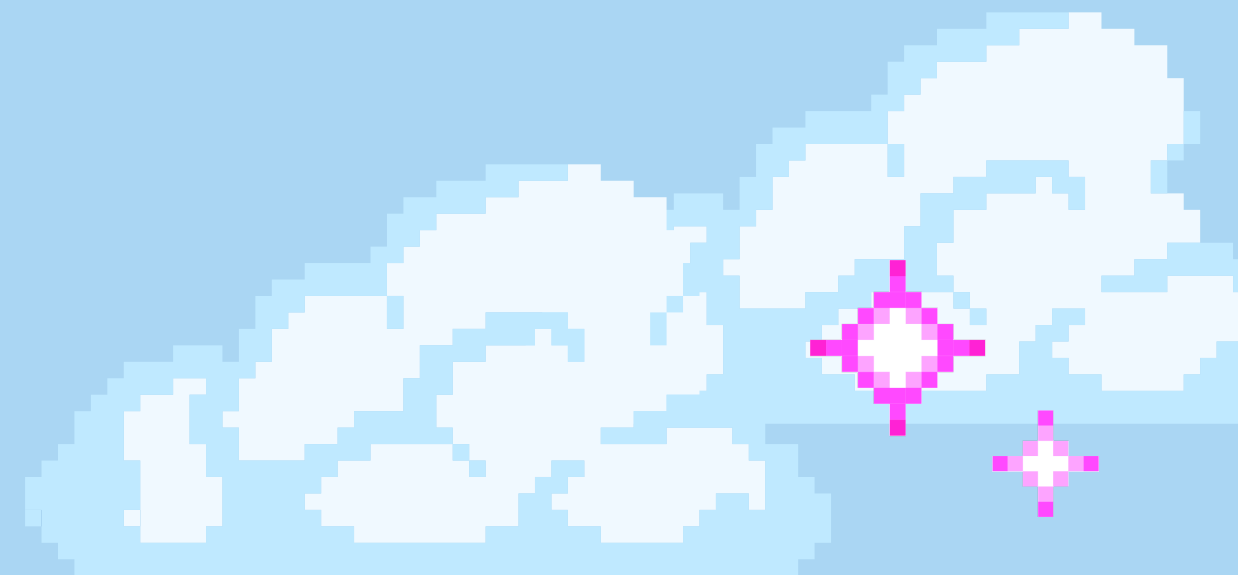


```
1 // calculate n1 and n2 with operator
2 int calculate(char *c, int n1, int n2)
3 {
4     switch (*c)
5     {
6         case '+': return (n1 + n2); break;
7         case '-': return (n1 - n2); break;
8         case 'x': return (n1 * n2); break;
9         case '/': return (n1 / n2); break;
10    }
11    return (0);
12 }
```

## FUNCTION CALCULATE

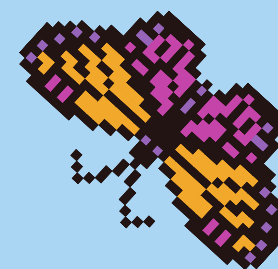
หลักการคำนวณค่า





```
1 void print_result(t_stack *stack, int value)
2 {
3     // if stack is empty
4     if (!stack)
5         printf("Empty Stack!\n");
6     // if stack is remaining data more than one
7     else if (stack->next)
8     {
9         printf("There is Remaining data in Stack!\n");
10        // traverse in linked-list
11        while (stack)
12        {
13            printf ("%d -> ", stack->data);
14            stack = stack->next;
15        }
16        printf ("Null\n");
17    }
18    // if stack can pop the element
19    else if (pop(&value, &stack))
20        printf("Value is: %d\n", value);
21 }
```

**FUNCTION** การแสดงผล  
**TRAVERSE** เพื่อเช็คค่าใน  
**STACK** และนำค่าไปเก็บไว้ใน **VALUE**



```

1 // argc is argument count
2 // argv is array of argument
3 int main(int argc, char **argv)
4 {
5     int i, n1, n2, value;
6     t_stack *stack;
7
8     i = 1;
9     stack = NULL;
10    // if argc == 1 it's mean no argument entered except "./a.out"
11    if (argc == 1)
12        printf("Please Enter a argument!!!\nExample:\n./a.out 3 4 5 x +\n");
13    else
14    {
15        while (i < argc)
16        {
17            if (!strchr("+-x/", argv[i][0]))
18            {
19                // check first of argument[i] is not + - x /
20                // then use atoi to check if string is numeric
21                if (!atoi(argv[i]) && strcmp(argv[i], "0"))
22                {
23                    printf("Please Enter only Numeric and + - x / only!\n");
24                    return (-1);
25                }
26                else if (!push(atoi(argv[i]), &stack))
27                    return (-1);
28            }
29            else if (strlen(argv[i]) == 1 && pop(&n2, &stack) && pop(&n1, &stack))
30            {
31                // then use atoi to convert string to int and push to stack
32                value = calculate(argv[i], n1, n2);
33                if (!push(value, &stack))
34                    return (-1);
35            }
36            else
37            {
38                printf("Error! Please check your Input.\n");
39                break;
40            }
41            i++;
42        }
43        print_result(stack, value);
44    }
45    return (0);
46 }

```

# MAIN





# ผล RUN ในแต่ละกรณี....

```
peakungg@PC-i5-11400: /mnt  ×  +  ∨  
peakungg postfix-calculator-stack-linkedlist $ ./a.out  
Please Enter a argument!!!  
Example:  
./a.out 3 4 5 x +  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 3 4 5 x +  
value is: 23  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 3 4 5 x  
there is Remaining data in Stack!  
20 → 3 → Null  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 3 4 5  
there is Remaining data in Stack!  
5 → 4 → 3 → Null  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 3 4 5 x + +  
Error! Please check your Input.  
Empty Stack!  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 3 4 5 x3  
Error! Please check your Input.  
there is Remaining data in Stack!  
5 → 4 → 3 → Null  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 3 2 x1 1  
Error! Please check your Input.  
there is Remaining data in Stack!  
2 → 3 → Null  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 1 +2 3 +  
Error! Please check your Input.  
value is: 1  
peakungg postfix-calculator-stack-linkedlist $ ./a.out 1 Hello word!  
Please Enter only Numeric and + - x / only!
```



# MEMEER!!!

นาย ฐปกร อิ่มอักษร

รหัสนิสิต 6430200159

นาย ชินวัตร สวัสดิ์

รหัสนิสิต 6430200817

นาย พงศกร ทิพย์สมเดช

รหัสนิสิต 6430200850

นางสาว ชมพูนุช สิงห์ทอง

รหัสนิสิต 6430200094

นางสาว ณ์ฐุชา สายเทพ

รหัสนิสิต 6430200205

นางสาว ภาคมน ดวงแก้วเลิศ

รหัสนิสิต 6430200531

คณะวิทยาศาสตร์ ศรีราชา สาขาวิทยาการคอมพิวเตอร์

START

MENU