System Software

Nehal Jhajharia (U20CS093)
Lab Assignment 9

Q) Write a program to construct LALR () parse table for the following grammar and check whether the given input can be accepted or not.

Grammar:

S -> AA A -> aA A -> b

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

#define MAX_STACK_SIZE 100
#define MAX_INPUT_SIZE 100

// Parse table for the grammar
int parse_table[2][3] = {
    {3, 4, -1},
    {-1, -1, 0}
};

// LR(0) items set
char* lr0_items[6] = {
    "S' -> .S",
    "S -> .AA",
    "A -> .aA",
    "A -> .b",
    "S -> AA.",
    "A -> aA."
};
```

```
// LR(1) items set
char* lr1_items[8][2] = {
    {"S' -> .S", "$"},
    {"S -> .AA", "ab"},
    {"A -> .aA", "a"},
    {"A -> .b", "ab"},
    {"S -> AA.", "$"},
    {"A -> aA.", "a"},
    {"A -> b.", "ab"},
    {"S' -> S.", "$"}
};

// Stack for the parser
int stack[MAX_STACK_SIZE];
int top = -1;

// Push an item onto the stack
void push(int state) {
    if (top >= MAX_STACK_SIZE - 1) {
        printf("Error: Stack overflow\n");
        exit(1);
    }
    stack[++top] = state;
}

// Pop an item from the stack
int pop() {
    if (top < 0) {
        printf("Error: Stack underflow\n");
        exit(1);
    }
    return stack[top--];
}

// Get the action for a given state and input symbol
int get_action(int state, char input) {
```

```
int symbol_index;
if (input == 'a') {
    symbol_index = 0;
} else if (input == 'b') {
    symbol_index = 1;
} else if (input == '$') {
    symbol_index = 2;
} else {
    printf("Error: Invalid input symbol\n");
    exit(1);
}
return parse_table[state][
```