# Lab 6: Recursive Descent Parser

Date: 20/02/2026

## Write a recursive descent parser for the following simple grammars.

**0. E --> TEprime**
**E' --> +TE' / ϵ**
**T --> FT'**
**T' --> *FT' / ϵ**
**F --> (E) / i**

Code:

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

int curr = 0;
char str[100];

void E();
void Eprime();
void T();
void Tprime();
void F();

void invalid() {
	printf("----------ERROR!----------\n");
	exit(0);
}
void valid() {
	printf("---------SUCCESS!---------\n");
	exit(0);
}

void E() {
	T();
	Eprime();
}
```

```
void Eprime() {
	if (str[curr] == '+') {
		curr++;
		T();
		Eprime();
	}
}
void T() {
	F();
	Tprime();
}
void Tprime() {
	if (str[curr] == '*') {
		curr++;
		F();
		Tprime();
	}
}
void F() {
	if (str[curr] == '(') {
		curr++;
		E();
		if (str[curr] == ')') {
			curr++;
			return;
		} else {
			invalid();
		}
	} else if (str[curr] == 'i') {
		curr++;
		return;
	} else {
		invalid();
	}
}

int main()
{
	printf("Enter string: ");
	scanf("%s", str);

	E();
```

```
        if (str[curr] == '$')
                valid();
        else
                invalid();

        return 0;
}
```

## Output:

```
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q0
Enter string: i+i*i$
---------SUCCESS!---------
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q0
Enter string: i*i+$
----------ERROR!----------
```

## 1. S→a | > | ( T )
## T→T, S|S

Code:

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

int curr = 0;
char str[100];

void S();
void T();
void Tprime();

void invalid() {
	printf("----------ERROR!----------\n");
	exit(0);
}
void valid() {
	printf("---------SUCCESS!---------\n");
	exit(0);
}

void S() {
	if (str[curr] == 'a') {
		curr++;
		return;
	} else if (str[curr] == '>') {
		curr++;
		return;
	} else if (str[curr] == '(') {
		curr++;
		T();
		if (str[curr] == ')') {
			curr++;
			return;
		} else {
			invalid();
		}
	} else {
```

```
                invalid();
        }
}
void T() {
        S();
        Tprime();
}
void Tprime() {
        if (str[curr] == ',') {
                curr++;
                S();
                Tprime();
        }
}

int main()
{
        printf("Enter string: ");
        scanf("%s", str);

        S();
        if (str[curr] == '$')
                valid();
        else
                invalid();

        return 0;
}
```

## Output:

```
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q1
Enter string: ((a,a),>)$
---------SUCCESS!---------
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q1
Enter string: (a,a),>$
----------ERROR!----------
```

**2.** **S→UVW**
**U →(S) | aSb | d**
**V → aV | ϵ**
**W → cW | ϵ**

Code:

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

int curr = 0;
char str[100];

void S();
void U();
void V();
void W();

void invalid() {
	printf("----------ERROR!----------\n");
	exit(0);
}
void valid() {
	printf("---------SUCCESS!---------\n");
	exit(0);
}

void S() {
	U();
	V();
	W();
}
void U() {
	if (str[curr] == '(') {
		curr++;
		S();
		if (str[curr] == ')') {
			curr++;
			return;
		} else {
			invalid();
```

```
                }
        } else if (str[curr] == 'a') {
                curr++;
                S();
                if (str[curr] == 'b') {
                        curr++;
                        return;
                } else {
                        invalid();
                }
        } else if (str[curr] == 'd') {
                curr++;
                return;
        } else {
                invalid();
        }
}
void V() {
        if (str[curr] == 'a') {
                curr++;
                V();
        }
}
void W() {
        if (str[curr] == 'c') {
                curr++;
                W();
        }
}

int main()
{
        printf("Enter string: ");
        scanf("%s", str);

        S();
        if (str[curr] == '$')
                valid();
        else
                invalid();

        return 0;
```

```
}
```

## Output:

```
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q2
Enter string: (adb)aaccc$
---------SUCCESS!---------
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q2
Enter string: (ab)ac$
----------ERROR!----------
```

**3. S → aAcBe**
**A → Ab|b**
**B → d**

Code:

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

int curr = 0;
char str[100];

void S();
void A();
void Aprime();
void B();

void invalid() {
	printf("----------ERROR!----------\n");
	exit(0);
}
void valid() {
	printf("---------SUCCESS!---------\n");
	exit(0);
}

void S() {
	if (str[curr] == 'a') {
		curr++;
		A();
		if (str[curr] == 'c') {
			curr++;
			B();
			if (str[curr] == 'e') {
				curr++;
				return;
			} else {
				invalid();
			}
		} else {
			invalid();
```

```
                }
        } else {
                invalid();
        }
}
void A() {
        if (str[curr] == 'b') {
                curr++;
                Aprime();
        } else {
                invalid();
        }
}
void Aprime() {
        if (str[curr] == 'b') {
                curr++;
                Aprime();
        }
}
void B() {
        if (str[curr] == 'd') {
                curr++;
                return;
        } else {
                invalid();
        }
}

int main()
{
        printf("Enter string: ");
        scanf("%s", str);

        S();
        if (str[curr] == '$')
                valid();
        else
                invalid();

        return 0;
}
```

## Output:

```
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q3
Enter string: abbbbcde$
---------SUCCESS!---------
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q3
Enter string: acde$
----------ERROR!----------
```

**4. S →(L) | a**
**L → L,S | S**

Code:

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

int curr = 0;
char str[100];

void S();
void L();
void Lprime();

void invalid() {
	printf("----------ERROR!----------\n");
	exit(0);
}
void valid() {
	printf("---------SUCCESS!---------\n");
	exit(0);
}

void S() {
	if (str[curr] == '(') {
		curr++;
		L();
		if (str[curr] == ')') {
			curr++;
			return;
		} else {
			invalid();
		}
	} else if (str[curr] == 'a') {
		curr++;
		return;
	} else {
		invalid();
	}
}
```

```
void L() {
	S();
	Lprime();
}
void Lprime() {
	if (str[curr] == ',') {
		curr++;
		S();
		Lprime();
	}
}

int main()
{
	printf("Enter string: ");
	scanf("%s", str);

	S();
	if (str[curr] == '$')
		valid();
	else
		invalid();

	return 0;
}
```

## Output:

```
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q4
Enter string: ((a,a),(a))$
---------SUCCESS!---------
6CSEC2@debian:~/Documents/230905252/Lab6$ ./exe/q4
Enter string: (a,a),$
----------ERROR!----------
```