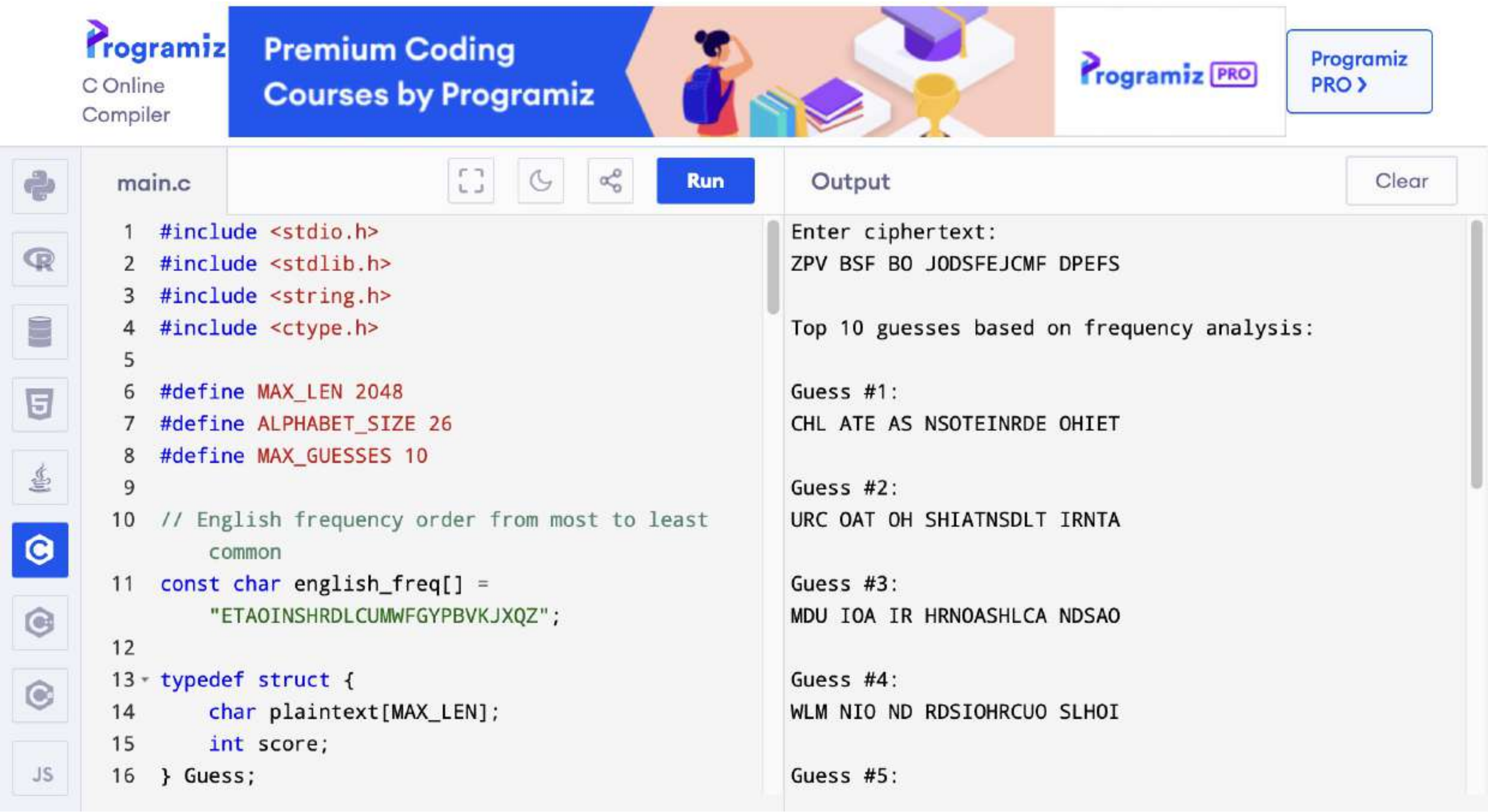
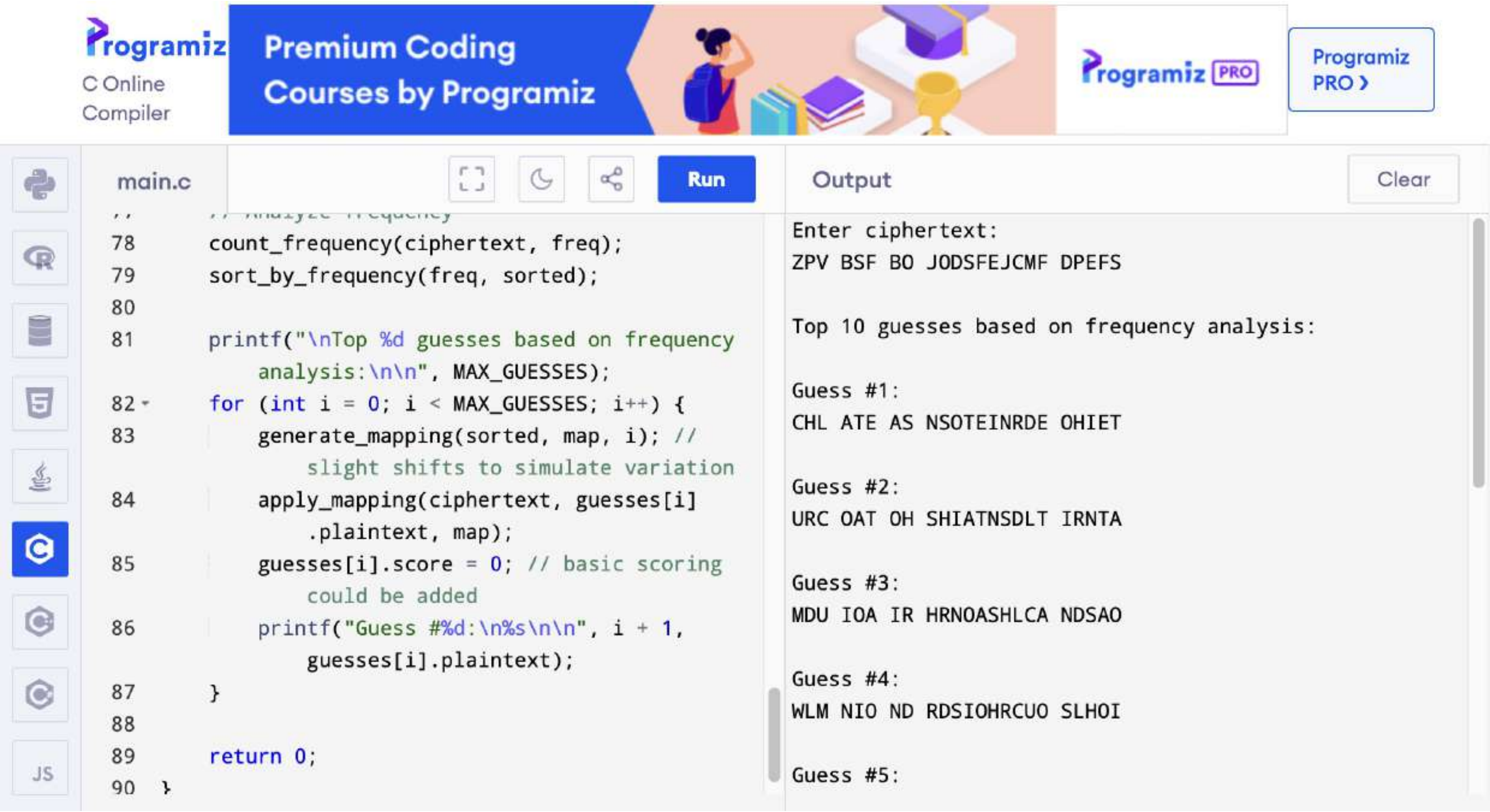


```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>

#define MAX_LEN 2048
#define ALPHABET_SIZE 26
#define MAX_GUESSES 10

// English frequency order from most to least common
const char english_freq[] = "ETAOINSHRDLCUMWFGYPBVKJXQZ";

typedef struct {
    char plaintext[MAX_LEN];
    int score;
} Guess;
```

```
Enter ciphertext:
ZPV BSF BO JODSFEJCMF DPEFS

Top 10 guesses based on frequency analysis:

Guess #1:
CHL ATE AS NSOTEINRDE OHIET

Guess #2:
URC OAT OH SHIATNSDLT IRNTA

Guess #3:
MDU IOA IR HRNOASHLCA NDSAO

Guess #4:
WLM NIO ND RDSIOHRCUO SLHOI

Guess #5:
```



```c
    // Analyze frequency
    count_frequency(ciphertext, freq);
    sort_by_frequency(freq, sorted);

    printf("\nTop %d guesses based on frequency analysis:\n\n", MAX_GUESSES);
    for (int i = 0; i < MAX_GUESSES; i++) {
        generate_mapping(sorted, map, i); // slight shifts to simulate variation
        apply_mapping(ciphertext, guesses[i].plaintext, map);
        guesses[i].score = 0; // basic scoring could be added
        printf("Guess #%d:\n%s\n\n", i + 1, guesses[i].plaintext);
    }

    return 0;
}
```

```
Enter ciphertext:
ZPV BSF BO JODSFEJCMF DPEFS

Top 10 guesses based on frequency analysis:

Guess #1:
CHL ATE AS NSOTEINRDE OHIET

Guess #2:
URC OAT OH SHIATNSDLT IRNTA

Guess #3:
MDU IOA IR HRNOASHLCA NDSAO

Guess #4:
WLM NIO ND RDSIOHRCUO SLHOI

Guess #5:
```