if-else: 2 choices

if (Expression-1)
    Statement-1
else
    Statement-2
Nextstatement
else-if with 3 choices

if (Expression-1)
    Statement-1
else if (Expression-2)
    Statement-2
else
    Statement-3
Nextstatement

else-if with 4 choices

if (Expression-1)
    Statement-1
else if (Expression-2)
    Statement-2
else if (Expression-3)
    Statement-3
else
    Statement-4
Nextstatement
Flowchart of `else-if`

Complex Conditions.
Example: Variable Rate 2
Example: Variable Rate 2 (1)

```c
#define FIRST_THRESHOLD 2500
#define SECOND_THRESHOLD 10000
#include <stdio.h>

void main() {
    float rate1, rate2, rate3, interest_rate; /* interest rates in percents */
    float capital; /* capital in dollars */
    float annual_interest; /* annual interest in dollars */

    /* read data */
    printf("Interest rates (%%xx.xx): ");
    scanf("%f %f %f", &rate1, &rate2, &rate3);
    printf("Capital ($$.cc): ");
    scanf("%f", &capital);

    /* calculate the rate */
    if (capital < FIRST_THRESHOLD)
        interest_rate = rate1;
    else if (capital < SECOND_THRESHOLD)
        interest_rate = rate2;
    else
        interest_rate = rate3;
    printf("The rate for $%.2f is %f\n", capital, interest_rate);

    /* calculate capital */
    annual_interest = capital * interest_rate / 100;
    printf("Interest %6.2f; Total %9.2f\n", annual_interest, capital + annual_interest);
}
```

Example: Variable Rate 2 (2)
Example: QPA

```c
#define MINQPA 2.75
#define MINISQPA 3.00
#include <stdio.h>

void main() {
    float qpa1, qpa2;
    /* read data */
    printf("Your general QPA?: "); scanf("%f", &qpa1);
    printf("Your IS QPA?: "); scanf("%f", &qpa2);

    /* make decision */
    if (qpa1 < MINQPA)
        printf("Your general QPA is too low for BSIS\n");
    else if (qpa2 < MINISQPA)
        printf("Your Information Science QPA is too low for BSIS\n");
    else /* here qpa1 >= MINQPA and qpa2 >= MINISQPA */
        printf("You are admitted to BSIS program!\n");
}
```

Complex Conditions: AND

- **AND operation** `ex1 && ex2`
  - evaluated to 1 (true) if each of ex1 and ex2 are not equal to 0 (each is true is C terms)
  - otherwise evaluated to 0
  
  ```c
c = 4;
c > 3 && c < 5 \Rightarrow true \Rightarrow 1
c < 3 && (c % 2 == 0) \Rightarrow false \Rightarrow 0
c > 3 && c < 9 && (c % 3 == 0) \Rightarrow false \Rightarrow 0
```
Complex Conditions: OR

- OR operation: \( \text{ex1} \lor \text{ex2} \)
  - evaluated to 1 (true) if at least one of \( \text{ex1} \) and \( \text{ex2} \) are not equal to 0 (at least one is true is C terms)
  - otherwise evaluated to 0

\[
\begin{align*}
\text{c} &= 4; \\
c \geq 3 \lor c < 5 &\Rightarrow \text{true} \Rightarrow 1 \\
c < 3 \lor (c \% 2 == 0) &\Rightarrow \text{true} \Rightarrow 1 \\
c < 3 \lor c > 9 \lor (c \% 3 == 0) &\Rightarrow \text{false} \Rightarrow 0 \\
c < 3 \lor c > 9 \lor (c \% 2 == 0) &\Rightarrow \text{true} \Rightarrow 1
\end{align*}
\]

Example: QPA again

```c
#define MINQPA 2.75
#define MINISQPA 3.00
#include <stdio.h>

void main() {
    float qpa1, qpa2;
    /* read data */
    printf("Your general QPA?: "); scanf("%f", &qpa1);
    printf("Your IS QPA?: "); scanf("%f", &qpa2);

    /* make decision */
    if (qpa1 < MINQPA || qpa2 < MINISQPA)
        printf("Your QPA is too low for BSIS\n");
    else /* here qpa1 >= MINQPA and qpa2 >= MINISQPA */
        printf("You are admitted to BSIS program!\n");
}
```
Encryption Engine

Calculating an Encrypted Char

- Calculate “number of a character” in its category:
  \[
  \text{charnum} = \text{ch} - 'a'; /* a => 0, b => 1... z => 25 */
  \]
- Calculate new “shifted” number of character:
  \[
  (\text{charnum} + \text{SHIFT}) \% \text{NLCHARS} /* should be 0 to 25 */
  \]
- Calculate the character corresponding to this new number (i.e., 0 is 'a', 1 is 'b', ... 25 is 'z'):
  \[
  \text{encrypted} = 'a' + (\text{charnum} + \text{SHIFT}) \% \text{NLCHARS}
  \]
Example: Encryption Engine

```c
#include <stdio.h>
#define SHIFT 1
#define NLCHARS ('z'-'a'+1) /* 26 */
#define NUCHARS ('Z'-'A'+1) /* 26 */

void main () {
    int ch, charnum;
    while((ch = getchar()) != EOF)
        if(ch >= 'a' && ch <= 'z') {
            charnum = ch - 'a'; /* a => 0, b => 1... z => 25 */
            putchar('a' + (charnum + SHIFT) % NLCHARS);
        } else if (ch >= 'A' && ch <= 'Z') {
            charnum = ch - 'A'; /* A => 0, B => 1... Z => 25 */
            putchar('A' + (charnum + SHIFT) % NUCHARS);
        } else
            putchar(ch);
}
```

Before next lecture:

- Do reading assignment
- Perry: Chapter 12
- Run Classroom Examples
- Check yourself by working with QuizQuide and WADEIn system
- Exercise: Count number of letters a, b and c in a text (similar to new line counting, but uses simple else-if)
- Assignment 7