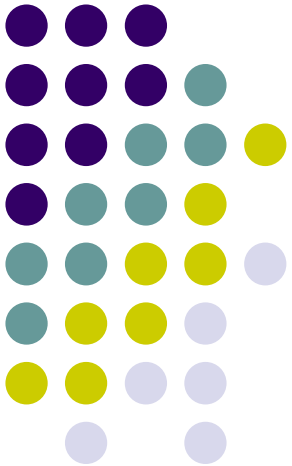
PDS Lab Section 15

Helper slides on if-else



Conditional Statements

- Allow different sets of instructions to be executed depending on truth or falsity of a logical condition
- Also called Branching
- How do we specify conditions?
  - Using expressions
    - non-zero value means condition is true
    - value 0 means condition is false
  - Usually logical expressions, but can be any expression
    - The value of the expression will be used

Branching: if Statement

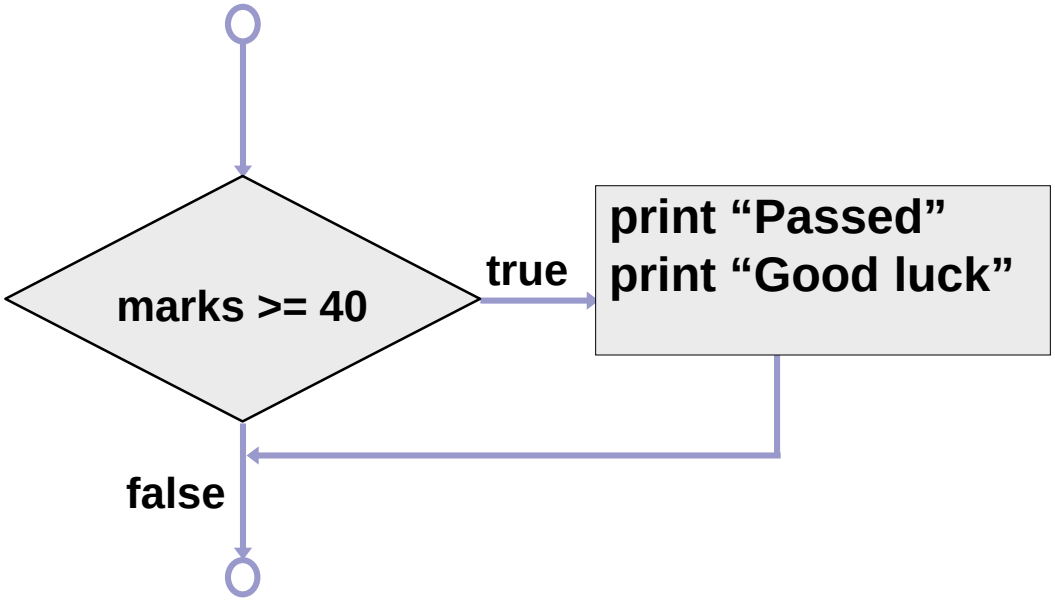
```
if (expression)
        statement;

if (expression) {
        Block of statements;
}
```

The condition to be tested is any expression enclosed in parentheses. The expression is evaluated, and if it is true, the statement or the block of statements inside { } is executed.

Expression types:

- Arithmetic: evaluates to true if ≠ 0, else evaluates to false
  - if (a – b) printf("a not equal to b\n");
- Relational:
  - if (a > b) printf("a is larger than b\n");
  - Other operators: <, ==, >=, <=, !=,…
- Boolean:
  - if (a > b && a > c) printf("a is greater than both b and c\n");
  - && (logical AND), || (logical OR)




```
if (marks >= 40)  {
   printf(“Passed \n”);
   printf(“Good luck\n”);
}
printf (“End\n”) ;
```

Branching: if-else Statement

```
if (expression) {
    Block of
    statements;
}
else {
    Block of
    statements;
}
```

```
if (expression) {
    Block of statements;
}
else if  (expression) {
    Block of statements;
}
else {
    Block of statements;
}
```

Grade Computation

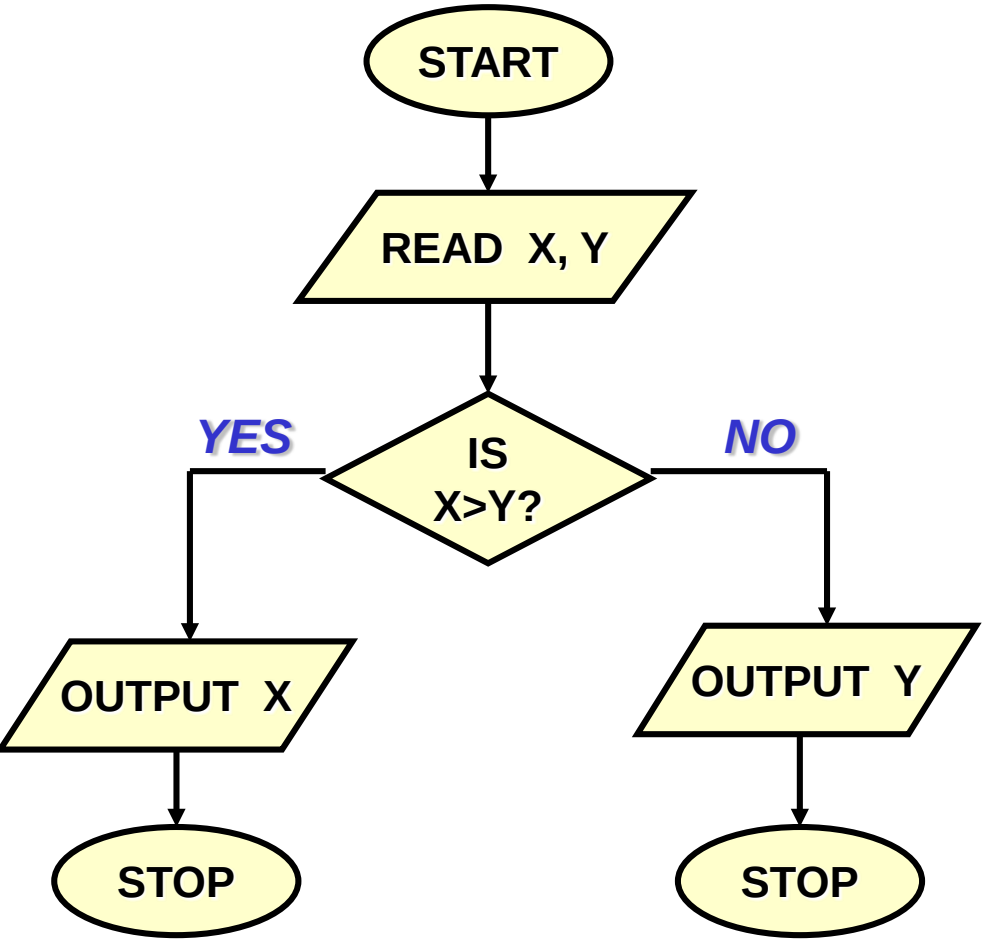
```
int main()  {
    int marks;
    scanf("%d", &marks);
    if (marks >= 80)
        printf ("A") ;
    else if (marks >= 70)
        printf ("B") ;
    else if (marks >= 60)
        printf ("C") ;
    else printf ("Failed");
    return 0;
}
```

Less than 60 is not fail in IIT, do not worry, just an example ☺

Same example, but print more than one thing (so give { } at the right places)

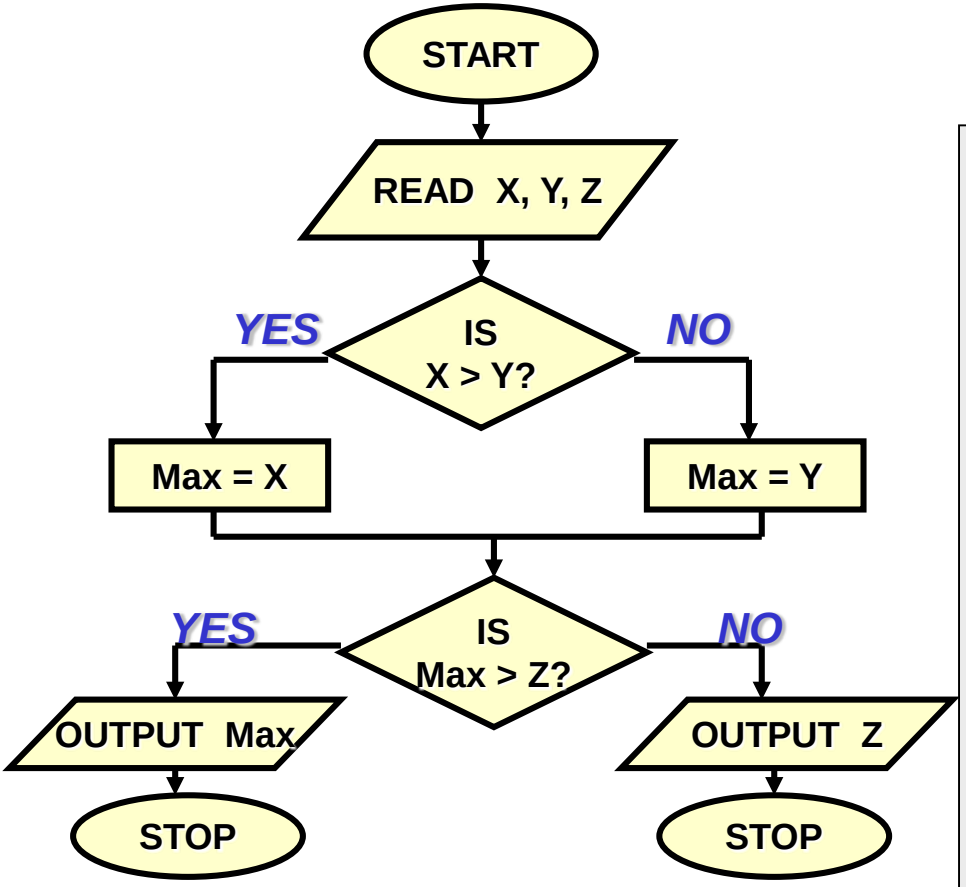
```
int main () {
      int marks;
     scanf (“%d”, &marks) ;
      if (marks>= 80) {
          printf (“A: ”) ;
          printf (“Good Job!”) ;
      }
      else if (marks >= 70)  printf (“B ”) ;
      else if (marks >= 60)  printf (“C ”) ;
      else {
          printf (“Failed: ”) ;
          printf (“Study hard for the supplementary”) ;
     }
      return 0;
}
```

Find the larger of two numbers




```
int main () {
    int x, y;
    scanf (“%d%d”, &x, &y);
    if (x > y)
        printf (“%d\n”, x);
    else
        printf (“%d\n”, y);
    return 0;
}
```

Find the largest of three numbers




```
int main () {
      int x, y, z, max;
      scanf ("%d%d%d",&x,&y,&z);
      if (x > y)
            max = x;
      else max = y;
      if (max > z)
            printf ("Max = %d", max) ;
      else printf ("Max = %d",z);
      return 0;
}
```

Another version

```
int main()  {
    int  a,b,c;
    scanf (“%d%d%d”, &a, &b, &c);
    if ((a >= b) && (a >= c))
        printf (“\n The largest number is: %d”, a);
    if ((b >= a) && (b >= c))
        printf (“\n The largest number is: %d”, b);
   if ((c >= a) && (c >= b))
        printf (“\n The largest number is: %d”, c);
   return 0;
}
```

Confusing Equality (==) and Assignment (=) Operators

- Dangerous error
  - Does not ordinarily cause syntax errors
  - Any expression that produces a value can be used in control structures
  - Nonzero values are true, zero values are false
- Example:

WRONG! Will always print the line

```
if ( payCode = 4 )
        printf( "You get a bonus!\n" );
```

Nesting of if-else Structures

- It is possible to nest if-else statements, one within another
- All “if” statements may not be having the “else” part
  - Confusion: which else matches with which if?
- Rule to be remembered:
  - An “else” clause is associated with the closest preceding unmatched “if”

Matching if's with else's

if (exp1) if (exp2) stmta else stmtb

Can be interpreted as either of the folloing two

```
if (exp1) {
    if (exp2)
        stmta
    else
        stmtb
}
```

OR

```
if (exp1) {
    if (exp2)
        stmta
}
else
    stmtb
```

?

Which one is the correct interpretation?

Give braces explicitly in your programs to match the else with the correct if to remove any ambiguity

Simple programs to try out

- Read in 3 integers (in 3 variables) and print the median
- Read in 3 floating point numbers (in 3 variables) and print them in ascending (increasing) order
- Read in 3 integers (in 3 variables) and print the difference between their maximum and minimum
- Repeat each of the above three with 5 numbers instead of 3
- Read in the center (x, y coordinates) and radius of a circle. Compute (store in a variable) and print its area and circumference. Then read in the x, y coordinates of a point p and check if the point p is inside the circle or not
- Read the equation of a straight line y = mx + c by reading m and c floating point). Then read in an integer k. Then find and print two points (x1, y1) and (x2, y2) on the line such that the distance between them is k.